

2561 #2
BT
01-24-02
PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant: Min-Koo Kim

Docket No: 678-758

Serial No: 09/986,236

Date: December 12, 2001

Filed: October 22, 2001

For: **HARQ DEVICE AND
METHOD FOR MOBILE
COMMUNICATION SYSTEM**

RECEIVED
JAN 15 2002
Technology Center 2600

Assistant Commissioner for Patents
Washington, D.C. 20231

TRANSMITTAL OF PRIORITY DOCUMENT

Sir:

Enclosed is a certified copy of Korean Appln. No. 62155/2000 filed
on October 21, 2000 from which priority is claimed under 35 U.S.C. §119.

Respectfully submitted,

Paul J. Farrell
Registration No. 33,494
Attorney for Applicant

DILWORTH & BARRESE, LLP
333 Earle Ovington Boulevard
Uniondale, New York 11553
(516) 228-8484

PJF:cm

CERTIFICATE OF MAILING UNDER 37 C.F.R. § 1.8 (a)

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail, postpaid in an envelope, addressed to the: Commissioner of Patents and Trademarks, Washington, D.C. 20231 on December 12, 2001

Dated: December 12, 2001

Paul J. Farrell



RECEIVED

JAN 15 2002

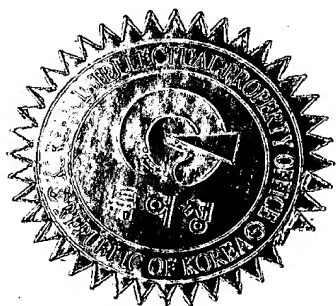
별첨 사본은 아래 출원의 원본과 동일함을 증명함. Technology Center 2600

This is to certify that the following application annexed hereto is a true copy from the records of the Korean Intellectual Property Office.

출원 번호 : 특허출원 2000년 제 62155 호
Application Number PATENT-2000-0062155

출원 년 월 일 : 2000년 10월 21일
Date of Application OCT 21, 2000

출원인 : 삼성전자 주식회사
Applicant(s) SAMSUNG ELECTRONICS CO., LTD.



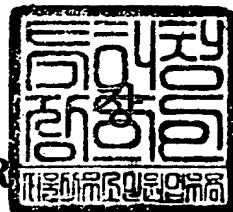
2001 년 10 월 22 일

특

허

청

COMMISSIONER



【서지사항】

【서류명】	특허출원서
【권리구분】	특허
【수신처】	특허청장
【참조번호】	0008
【제출일자】	2000.10.21
【국제특허분류】	H04M
【발명의 명칭】	복합 재전송형식을 사용하는 데이터 통신시스템의 데이터 송수신장치 및 방법
【발명의 영문명칭】	Hybrid Automatic Repeat Request Method and Apparatus for Improving Data Communication Systems
【출원인】	
【명칭】	삼성전자 주식회사
【출원인코드】	1-1998-104271-3
【대리인】	
【성명】	이건주
【대리인코드】	9-1998-000339-8
【포괄위임등록번호】	1999-006038-0
【발명자】	
【성명의 국문표기】	김민구
【성명의 영문표기】	KIM,Min-Koo
【주민등록번호】	640820-1067025
【우편번호】	442-470
【주소】	경기도 수원시 팔달구 영통동 973-3 우성아파트 822-406
【국적】	KR
【취지】	특허법 제42조의 규정에 의하여 위와 같이 출원합 니다. 대리인 이건주 (인)

【수수료】

【기본출원료】	20	면	29,000	원
---------	----	---	--------	---

【가산출원료】	41	면	41,000	원
---------	----	---	--------	---

【우선권주장료】	0	건	0	원
----------	---	---	---	---

【심사청구료】	0	항	0	원
---------	---	---	---	---

【합계】	70,000	원		
------	--------	---	--	--

【첨부서류】	1. 요약서·명세서(도면)_1통			
--------	-------------------	--	--	--

【요약서】**【요약】**

본 발명은 무선 통신시스템의 송신장치에서 물리 계층 패킷을 소정 부호화 절차에 의해 서브 부호 셋을 생성하여 패킷 부호 결합방식에 의한 전송방법에 관한 것으로, 서브 부호 셋을 소정 단위로 분할한 복수의 전송 프레임들을 순차적으로 전송하여 재 전송 요구를 수신하면 복수의 전송 프레임들의 전송을 완료한 후 상기 복수의 전송 프레임들을 순차적으로 재 전송하며, 수신장치는 송신장치로부터 최초 전송하고자 하는 PLP에 따른 전송 프레임들에 대해 패킷 코드 컴바이닝을 수행하고, 재 전송에 따른 전송 프레임들 및 그 이후에 수신되는 전송 프레임들에 대해 다이버시티 컴바이닝을 수행함으로써 복합재전송을 사용하는 통신시스템의 시스템 성능을 향상시킬 수 있다.

【대표도】

도 10

【색인어】

HARQ, TURBO CODES, PACKET COMBINING, SOFT COMBINING, THROUGHPUT, CODE COMBINING

【명세서】**【발명의 명칭】**

복합 재전송형식을 사용하는 데이터 통신시스템의 데이터 송수신장치 및
방법{Hybrid Automatic Repeat Request Method and Apparatus for Improving Data
Communication Systems}

【도면의 간단한 설명】

도 1은 터보부호를 사용하는 패킷 통신시스템에서 패킷 부호 결합과 패킷
다이버시티 결합에 따른 성능 차이를 보여주고 있는 도면.

도 2는 본 발명의 실시 예에 따른 구체적인 터보 부호화기의 구조를 보여주
고 있는 도면.

도 3은 본 발명의 실시 예에 따른 복합 재전송 방식을 적용한 패킷 처리 흐
름의 일 예를 보여주고 있는 도면.

도 4는 본 발명의 실시 예에 따른 복합 재전송 방식을 적용한 패킷 처리 흐
름의 다른 예를 보여주고 있는 도면.

도 5는 도 4에 의해 순방향 및 역방향으로 전송되는 슬롯들의 관계를 보여
주고 있는 도면.

도 6은 도 5에 의해 순방향 및 역방향으로 전송되는 슬롯들의 관계를 보여
주고 있는 도면.

도 7은 본 발명의 실시 예에 따른 터보 보완 부호에 따른 서브 부호를 생성하기 위한 절차를 도시하는 도면.

도 8은 본 발명의 실시 예에 따른 터보 보완 부호에 따른 서브 부호를 생성하기 위한 절차를 도시하는 도면.

도 9는 본 발명의 실시 예에 따른 터보 보완 부호에 따른 서브 부호를 생성하기 위한 절차를 도시하는 도면.

도 10은 본 발명의 실시 예에 따른 복합재전송방식을 사용하는 데이터 통신시스템에서의 데이터 송신을 위한 제어 흐름을 보여주고 있는 도면.

도 11 및 도 12는 본 발명이 실시 예에 따른 복합 재전송방식을 사용하는 데이터 통신시스템에서의 데이터 수신을 위한 제어 흐름을 보여주고 있는 도면.

【발명의 상세한 설명】

【발명의 목적】

【발명이 속하는 기술분야 및 그 분야의 종래기술】

<12> 본 발명은 무선 통신시스템에서 데이터 전송장치 및 방법에 관한 것으로, 특히 데이터를 전송하는 중에 오류가 발생한 데이터의 재전송을 처리하는 장치 및 방법에 관한 것이다.

<13> 통상적으로 무선 통신시스템에서는 순방향 데이터 통신을 수행하는 경우, 이동국은 기지국으로부터 순방향(forward) 채널 즉, 전용채널(DCH: Dedicated Channel) 등과 같은 채널을 할당받아 데이터 패킷을 수신하게 된다. 상기 무선

통신시스템은 위성시스템, ISDN, 디지털 셀룰러(Digital cellular), W-CDMA, UMTS, IMT-2000 등을 통칭한다. 이 때, 이동국은 성공적으로 수신되어진 패킷은 상위 계층으로 전달한다. 하지만, 오류가 발생한 패킷에 대해서는 복합 재전송방식(HARQ: Hybrid Automatic Repeat Request)을 사용하여 재전송을 요청하게 된다. 상기 복합 재전송방식이란 오류 정정 부호(FEC: Forward Error Correction)와 오류검출 시에 데이터 패킷의 재전송을 요구하는 재전송방식(ARQ: Automatic Repeat Request)을 모두 사용하는 재전송방식이다. 이는 오류 검출시 채널 부호화방식(Channel coding scheme)을 사용하여 데이터 전송효율성, 즉 이득율(throughput)을 높이고 시스템의 성능을 개선하기 위한 전송방식이다. 상기 무선 통신시스템에서 주로 사용되는 채널 부호화방식으로는 컨벌루션널 부호(convolutional codes)나 터보 부호(turbo codes) 등을 FEC로 하는 채널 부호화방식이 사용되었다.

<14> 또한, 하이브리드 ARQ(Hybrid ARQ)방식을 사용하는 시스템에서 전송효율(throughput)을 개선하고자 연성 결합(Soft combining)을 사용하며 여기에는 두 가지 방식이 가능하다. 첫 번째 방식으로 packet diversity combining(패킷 다이버시티 결합)을 사용하는 구조이며, 두번째 방식으로 packet code combining(패킷 부호 결합)을 사용하는 방식이다. 일반적으로 이 두 방식을 모두 Soft Packet Combining(연성 패킷 결합)이라고 말하며 이 중 상기 packet diversity combining 방식은 상기 packet code combining 방식에 비하여 성능면에서 sub-optimal 방식이지만 구현의 편의성 때문에 성능상의 큰 손실이 없는 경우 자주 사용되는 방식이다.

<15> 통상적으로, Packet(패킷)을 전송하는 시스템에서 전송효율 (Throughput)을 증가시키기 위해서 연성 패킷 부호 결합을 사용한다. 즉, 각각의 전송마다 전달되는 Packet에 대하여 부호율이 R인 상호 다른 부호를 전송하고 복호된 결과 수신된 패킷에 오류가 검출되는 경우 이를 소멸시키지 않고 저장한 뒤에 향후 재전송되어 올 패킷과 연성결합 (Soft Combining)을 하는 방식을 말한다. 이때 재전송되는 패킷은 상호 다른 부호가 사용될 수 있다. 즉, 상기 패킷 부호 결합 방식은 부호율이 R인 N개의 패킷을 수신한 경우 각각의 패킷을 사용하여 실효 부호율(effective code rate)이 R/N 인 부호로 전환한 뒤에 복호함으로써 부호화이득(coding gain)을 얻는 효과를 갖는 방식이다.

<16> 반면에 패킷 다이버시티 결합은 각각의 전송마다 전달되는 Packet에 대하여 부호율이 R인 동일한 부호를 전송하고 복호된 결과 수신된 패킷에 오류가 검출되는 경우 이를 소멸시키지 않고 저장한 뒤에 향후 재전송되어 올 패킷과 연성결합(Soft Combining)을 하는 방식을 말한다. 이때 재전송되는 패킷은 항상 동일한 부호가 사용된다. 따라서 패킷 다이버시티 결합은 랜덤채널에서 일종의 Symbol Averaging (심볼에너지 평균과정)으로 볼 수 있으며 수신심볼 연성출력(soft output)을 평균함으로써 얻는 잡음전력감소 효과와 페이딩 채널에서 복수개의 심볼을 전송함으로써 다중성 채널에서 제공되는 다중성이득(diversity gain)만을 사용하는 방식이라고 볼 수 있다. 이에 반해서 패킷 부호 결합은 이러한 이득 이외에도 Code structure(코드 구조)에 따른 추가의 Coding Gain(코딩 게인)을 가지고 있다.

<17> 현재까지의 패킷통신 시스템에서는 구현의 용이함 때문에 대부분 패킷 다이버시티 결합을 사용하고 있으며 동기방식의 IS-2000 시스템과 비동기방식의 UMTS 시스템 등에서 이러한 방식이 고려되고 있다. 그러나 이는 기존의 패킷통신 시스템들이 대부분 길쌈부호를 사용하였고, 길쌈부호의 경우 부호율 R이 낮은 부호(codes)를 사용하는 경우에 상기 패킷 다이버시티 결합을 사용한다 해도 그리 큰 이득이 제공되지 않기 때문이었다. 즉, $R=1/3$ 사용하는 시스템에서 재전송이 가능한 경우, 상기 패킷 다이버시티 결합을 사용하는 경우와 패킷 부호 결합을 사용하여 $1/6$ 부호율을 사용하는 경우의 성능차이가 그리 크지 않기 때문에 구현의 복잡도를 고려하여 패킷 다이버시티 결합을 사용하였다. 그러나 오류정정부호(Forward Error Correction Codes: FEC)로 터보 부호를 사용하는 경우에는 이러한 기존의 개념과는 다른 방식이 요구된다. 왜냐하면 터보 부호는 반복 복호(iterative decoding)에 의해서 그 성능이 Shannon limit에 근접하도록 설계된 오류정정부호이며, 부호율에 따른 성능의 차이가 기존의 길쌈 부호(Convolutional codes)와는 달리 분명하게 존재하기 때문이다. 즉, 이러한 점을 고려할때 재전송을 사용하는 패킷통신 시스템에서는 터보 부호(Turbo codes)를 사용하는 패킷 부호 결합 방식을 구현하는 것이 성능을 향상시키는 방법이 된다.

【발명이 이루고자 하는 기술적 과제】

<18> 따라서 본 발명의 목적은 최적의 패킷 부호 결합(Packet Code Combining)을 위한 부호를 설계하기 위한 방법을 제공함에 있다.

<19> 본 발명의 다른 목적은 최적의 패킷 부호 결합(Packet Code Combining)을 위한 부호를 사용하여 패킷 부호 결합과 패킷 다이버시티 결합을 전송율에 따라 가변적으로 사용하는 시스템을 제공함에 있다.

<20> 본 발명의 또 다른 목적은 패킷 부호 결합과 패킷 다이버시티 결합을 전송율에 따라 가변적으로 사용하는 시스템을 운영하기 위한 복합재전송 방식의 프로토콜을 제공함에 있다.

【발명의 구성 및 작용】

<21> 이하 본 발명의 바람직한 실시 예를 첨부된 도면의 참조와 함께 상세히 설명한다.

<22> 우선 각 도면의 구성요소들에 참조부호를 부가함에 있어서, 동일한 구성요소들에 한해서는 비록 다른 도면상에 표시되더라도 가능한 동일 부호를 가지도록 하였다. 또한 본 발명을 설명함에 있어서, 관련된 공지기능 혹은 구성에 대한 구체적인 설명이 본 발명의 요지를 불필요하게 흐릴 수 있다고 판단된 경우 그 상세한 설명은 생략한다.

<23> 1. Packet Code Combining

<24> 패킷을 전송하는 시스템에서 데이터 전송 효율성, 즉 이득율(Throughput)을 증가시키기 위해서 연성 패킷 부호 결합(Soft Packet Code Combining)을 사용한다. 즉, 각각의 전송마다 전달되는 패킷에 대하여 부호율이 R인 상호 다른 부호

를 전송하고, 복호된 결과 수신된 패킷에 오류가 검출되는 경우 이를 소멸시키지 않고 저장한 뒤에 향후 재 전송되어 올 패킷과 연성결합(Soft Combining)을 하는 방식을 말한다. 이때 재 전송되는 패킷은 상호 다른 부호가 사용될 수 있다. 즉, 부호율이 R인 N개의 패킷을 수신한 경우 각각의 패킷을 사용하여 실효 부호율(effective code rate)이 R/N 인 부호로 전환한 뒤에 복호함으로써 부호화 이득(coding gain)을 얻는 효과를 주는 방식이 패킷 부호 결합(Packet Code Combining)이다.

<25> 이에 반면에 패킷 다이버시티 결합(Packet Diversity Combining)은 각각의 전송마다 전달되는 패킷에 대하여 부호율이 R인 동일한 부호를 전송하고, 복호된 결과 수신된 패킷에 오류가 검출되는 경우 이를 소멸시키지 않고 저장한 뒤에 향후 재 전송되어 올 패킷과 연성결합(Soft Combining)을 하는 방식을 말한다. 이때 재 전송되는 패킷은 항상 동일한 부호가 사용된다. 따라서, 상기 패킷 다이버시티 결합(Packet Diversity Combining)은 랜덤채널에서 일종의 심볼 에너지 평균과정(Symbol Averaging)으로 볼 수 있다. 또한, 수신심볼 연성출력(soft output)을 평균함으로써 얻는 잡음전력 감소 효과와 페이딩 채널에서 복수개의 심볼을 전송함으로써 다중성 채널에서 제공되는 다중성 이득(diversity gain)만을 사용하는 방식이라고 볼 수 있다. 하지만, 상기 패킷 부호 결합(Packet Code Combining)은 이러한 이득 이외에도 부호 구조(Code structure)에 따른 추가의 부호화 이득(Coding Gain)을 가지고 있다.

<26> 현재까지의 패킷 통신시스템에서는 구현의 용이함 때문에 대부분 패킷 다이버시티 결합(Packet Diversity Combining)을 사용하고 있으며, 동기방식의

IS-2000 시스템과 비동기방식의 UMTS 시스템 등에서 이러한 방식이 고려되고 있다. 그 이유는 기존의 패킷 통신시스템들이 대부분 컨벌루션널 부호(Convolutional codes)를 사용하였을 뿐만 아니라 상기 컨벌루션널 부호(Convolutional codes)의 경우 부호율 R이 낮은 부호를 사용하는 경우에 그리 큰 이득이 제공되지 않기 때문이었다. 즉, R=1/3 사용하는 시스템에서 재전송이 가능한 패킷 다이버시티 결합(Packet Diversity Combining)을 사용하는 경우와 패킷 부호 결합(Packet Code Combining)을 사용하여 1/6 부호율(code rate)을 사용하는 경우의 성능 차이가 그리 크지 않기 때문에 구현의 복잡도를 고려하여 패킷 다이버시티 결합(Packet Diversity Combining)을 사용하였다.

<27> 그러나 오류정정부호(Forward Error Correction Codes: FEC)로 터보 부호(Turbo Codes)를 사용하는 경우에는 이러한 기존의 개념과는 다른 방식이 요구된다. 왜냐하면 상기 터보 부호(Turbo codes)는 반복 복호(iterative decoding)에 의해서 그 성능이 Shannon limit에 근접하도록 설계된 FEC이며, 부호율에 따른 성능의 차이가 기존의 컨벌루션널 부호(Convolutional codes)와는 달리 분명하게 존재하기 때문이다. 즉, 이 점을 근간으로 터보부호를 사용하는 패킷 부호 결합(Packet Code Combining) 방식을 구현하는 것이 재전송을 포함하는 패킷통신 시스템에서 성능을 향상시키는 방법이 된다.

<28> 따라서 후술되어질 본 발명의 상세한 설명에서는 상기의 분석 결과에 근거해서 최적의 패킷 부호 결합(Packet Code Combining)을 위한 부호를 설계하기 위한 방법을 제시할 것이다. 또한 이러한 부호를 사용하여 패킷 부호 결합(Packet Code Combining)과 패킷 다이버시티 결합(Packet Diversity Combining)을 전송을

에 따라서 가변적으로 사용하는 시스템을 제안하고, 이러한 시스템을 운영하기 위한 복합재전송 방식의 프로토콜에 대해 설명할 것이다.

<29> 본 발명에서 제안하는 패킷 부호 결합(Packet code combining)과 패킷 다이버시티 결합(packet diversity combining)을 전송율에 따라서 가변적으로 사용하는 시스템의 동작방식은 다음과 같다.

<30> 예를 들어 $R=1/5$ 터보부호를 사용하는 시스템의 경우 재 전송된 패킷들로부터 연성 결합(soft combining)에 의해 구한 부호어(code word)의 전체 부호율(overall code rate)이 $1/5$ 이 되기까지는 패킷 부호 결합(packet code combining)을 사용한다. 이후부터 재 전송되는 패킷들에 대해서는 우선 패킷 다이버시티 결합(packet diversity combining)을 사용하고, 이어서 패킷 부호 결합(packet code combining)을 사용한다. 즉, 처음 전송하는 패킷의 부호율 R 이 $1/3$ 인 경우 다음 번 재전송 요구 때에는 전체 부호율(Overall code rate) R 이 $1/5$ 이 되도록 나머지 리던던시(redundancy)를 전송한다. 따라서 수신기가 두개의 패킷들을 모두 수신하면 전체 부호율(overall code rate)은 $R=1/5$ 이 되며, 그 이후에 전송되는 패킷들에 대해서는 각각을 반복해서 전송한다. 한편, 수신기는 패킷 다이버시티 결합(packet diversity combining)을 사용한 후에 $R=1/5$ 부호율(code rate)을 기준으로 재전송 패킷을 패킷 부호 결합(packet code combining)을 수행한다.

<31> 일반적으로 컨벌루션널 부호(convolutional codes)의 경우에 부호율(code rate)이 작은 경우 패킷 다이버시티 결합(packet diversity combining)과 패킷 부호 결합(packet code combining) 사이에 성능 차이가 그리 크지 않은 것으로

알려져 있다. 그러나, 터보부호의 경우에는 컨벌루션널 부호와는 달리 그 차이가 매우 크며, 패킷 부호 결합(packet code combining)이 패킷 다이버시티 결합(packet diversity combining)에 비하여 상당한 성능 이득이 제공된다. 따라서 이러한 성질을 터보부호를 사용하는 Hybrid ARQ II/III에 적용하여 상당한 이득율(throughput)의 개선을 얻을 수 있다.

<32> 도 1은 터보부호의 경우 패킷 부호 결합(packet code combining) 방식과 패킷 다이버시티 결합(packet diversity combining) 방식 사이의 성능 차이를 보여주고 있는 도면이다. 일반적으로 동일한 부호 심볼 에너지(coded symbol energy) E_s 와 동일한 부호율(code rate)을 가정할 때 컨벌루션널 부호(convolutional codes)와는 달리 터보부호(turbo codes)의 성능(performance)은 반복(iteration)이 충분히 제공되는 경우 Shannon Channel Capacity limit에 근접하는 성능을 부호율(code rates)에 따라서 제공한다. 따라서 동일한 E_s 를 가정하는 경우 낮은 부호율(low code rate)의 터보부호(turbo codes)는 높은 부호율(high code rate)의 터보부호에 비하여 상당한 성능 이득을 제공하는 것으로 알려져 있다. 예를 들어 부호율이 $R=1/3$ 에서 $R=1/6$ 로 감소하는 경우 Shannon Channel Capacity limit의 변화를 분석해서 그 성능 차이를 예측할 수 있다. 여기서 주의할 부분은 HARQ를 사용하지 않는, 즉 재전송을 사용하지 않는 시스템의 경우 기존의 부호율(code rates)에 따른 터보부호의 성능 분석이 부호율 R 의 감소로 인한 심볼 에너지(symbol energy)의 감소를 고려해서 비교한 반면에 HARQ를 사용하는 시스템에서는 재전송마다 동일한 심볼 에너지(symbol energy) E_s 를 사용하므로 $R=1/3$ 의 경우나 $R=1/6$ 의 경우나 동일한

심볼 에너지(symbol energy) E_s 를 가진다. 따라서, $R=1/3$ 의 부호를 2번 반복해서 패킷 다이버시티 결합(packet diversity combining)하는 경우는 AWGN(Additive White Gaussian Noise) 채널에서 E_s/N_0 면에서 최대 3dB의 이득을 가진다. 또한, $R=1/6$ 부호를 사용하는 경우에도 기존의 $R=1/6$ 부호의 심볼 에너지에 비하여 E_s/N_0 면에서 3dB의 이득을 가지므로 이를 고려하여 비교해야 한다. 따라서 일반적으로 부호율에 따른 부호의 성능 비교에서 사용하는 E_b/N_0 를 기준으로 그린 터보부호의 성능 곡선에서 $R=1/3$ 의 성능 곡선도 패킷 다이버시티 결합 이득(packet diversity combining gain)으로 +3dB 스케일(scale)이 평행 이동되고, $R=1/6$ 터보부호의 성능 곡선도 $R=1/3$ 과 동일한 심볼 에너지를 사용하는 가정에서 +3dB 스케일(scale)이 평행 이동되므로 결국 E_b/N_0 를 기준으로 그린 $R=1/3$ 터보부호와 $R=1/6$ 터보부호와의 성능 차이가 패킷 부호 결합(packet code combining)이 제공하는 성능 차이가 된다. 이러한 부호율에 따른 성능 차이는 다음의 Shannon Channel Capacity limit에 의해서 예측 가능하며, 최소의 성능 차이는 다음의 minimum required S/N을 사용하여 구할 수 있다.

<33> 2. Minimum Required E_b/N_0 for code rates

<34> 부호율이 R 이고, 복호기의 크기가 매우 큰 터보부호를 사용하는 시스템에서 오류가 전혀 발생하지 않는 에러 프리(error free) 채널을 제공하기 위한 최소 E_b/N_0 는 하기의 <수학식 1>에 의해 결정되어 질 수 있다.

<35> 【수학식 1】 $E_b/N_0 > (4R-1) / 2R$

<36> 상기 <수학식 1>에 근거하여 AWGN에서 각각의 부호율에 터보부호에서 요구되는 최소의 E_b/N_0 즉, minimum required E_b/N_0 를 하기의 <표 1>에 보였다.

<37> 【표 1】

Code rates	Required E_b/N_0 (dB)	Typical E_b/N_0 (db) for BER= 10^{-5}
3/4	0.86	
2/3	0.57	
1/2	0.00	
3/8	-0.414	
1/3	-0.55	
1/4	-0.82	
1/5	-0.975	
:		
0	-1.62	NA

<38> 상기 <표 1>에서의 Typical E_b/N_0 는 터보부호의 복호기의 크기 L 이 1024일 때 Bit error rate(BER)가 0.00001보다 작게 하기 위해서 요구되는 E_b/N_0 를 도시한 것이다. 상기 <표 1>에서 보여지고 있는 바와 같이 부호율 R 이 3/4, 2/3, 1/2, 3/8, 1/3, 1/4, 1/5, 1/6인 경우에 각각에 요구되는 E_b/N_0 는 각각 0.86, 0.57, 0.00, -0.414, -0.55, -0.82, -0.975, -1.084(dB)임을 알 수 있다. 또한, 상기 <표 1>에서 부호율 $R=1/3$ 인 부호를 사용하는 시스템과 부호율 $R=1/6$ 인 부호를 사용하는 시스템 사이에는 최소한 0.53dB 이상의 성능 차이가 있음을 알 수 있다. 이는 Shannon limit에 따른 최소의 성능 차이로 실제 구현 가능한 복호기와 시스템 환경을 고려하면 그 차이는 이 보다 증가한다. 실제로 시뮬레이션에 의한 성능 차이에 따르면 $R=2/3$ 부호를 사용하고 패킷 부호 결합(packet code combining)을 사용하는 시스템과, 동일한 부호율 $R=2/3$ 를 사용하고 패킷 다이버시티 결합(packet diversity combining)을 사용하는 시스템 사이에는 약 1.12dB

의 성능 차이가 있음을 알 수 있다. 하기의 <표 2>에서는 서브 부호율(sub code rate)로 $R=2/3$ 을 사용하는 시스템에서 한번의 재전송을 사용한 경우에 패킷 부호 결합(packet code combining)을 사용하는 시스템과 동일한 부호율 $R=2/3$ 를 사용하고 패킷 다이버시티 결합(packet diversity combining)을 사용하는 시스템 사이의 성능 차이를 보였다.

<39> 【표 2】

Items	Packet Combining	Code Combining
Mother code rate R_m	1/3 (X,Yo,Y'o) in Figure 1	1/3 (X,Yo,Y'o) in Figure 1
Block size (L)	496	496
Maximum number of iterations	8	8
Number of transmissions	2	2
Actual Tx code rate R_e for each transmission	2/3(by puncturing) See section 3.	2/3(by puncturing) See section 3.
Redundancy selection	Identical pattern for all transmissions. See section 3.	Different pattern for all transmissions. See section 3.
Soft Combining	Packet diversity combining	Packet code combining
Gain through retransmissions	Symbol repetition gain	Coding gain for low rate codes
Minimum required E_b/N_o in Table 3	+0.57(dB)	R-2/3 +0.57(dB) R-2/6 -0.55(dB)
Required E_b/N_o at 2'nd retransmissions	+0.57-3.0(dB)	-0.55-3.0(dB)
Relative performance gain	0	1.12(=0.57+0.55) dB
Simulated relative gain (@ BER= 10^{-5})	0	2.5(dB)

<40> 상기 <표 2>에서 보듯이 최소의 성능 차이가 1.12dB이므로 터보부호를 사용하는 시스템에서 패킷 부호 결합이 많은 성능 이득을 가지는 것을 알 수 있다. 따라서 상기의 분석에 의해서 터보부호를 사용하는 재전송 시스템에서 각각의 전송에 사용할 서브 부호율을 구하고, 이를 패킷 부호 결합(Packet Code Combining)을 위한 서브 부호 셋(sub code set)으로 사용하면 앞서 제시한 패킷

부호 결합에 의한 이득을 얻을 수 있다. 즉, 아래의 절차에 의해 구한 이러한 서브 부호 집합(sub codes set)을 사용함으로써 동일한 양의 재전송을 요구하는 시스템의 성능을 최대로 개선시킬 수 있다.

<41> 이하 설명되는 도 7 내지 도 9는 본 발명의 실시 예에 따른 터보 보완 부호에 따른 서브 부호들을 생성하기 위한 절차를 도시하고 있다. 특히, 도 7은 S개의 서브 부호들 중 첫 번째 서브부호(C_0)를 생성하기 위한 절차이고, 상기 도 8은 $C_1 \sim C_{s-2}$ 를 생성하기 위한 절차이며, 상기 도 9는 마지막 서브부호(C_{s-1})을 생성하기 위한 절차를 나타낸다.

<42> 한편, 이하 설명되는 ENC1(이하 제1부분이라 칭함)은 각 서브 부호 매트릭스를 두 부분으로 구분하여 상위의 정보부분(systematic part)과 제1패러티부분(1st parity party)을 나타내고, ENC2(이하 제2부분이라 칭함)는 하위의 제2패러티 부분(2nd parity part)을 나타낸다. 예를 들어, 도 2를 참조하면, 상기 정보어부분은 'x'이고, 상기 제1패러티부분은 'y0', 'y'1' 이며, 상기 제2패러티부분은 'y'0', 'y'1'이다.

<43> 상기 도 7을 참조하면, 먼저 401단계에서 데이터 전송시스템이 제공할 수 있는 최대의 부호율 R_{max} 를 정한다. 이는 사용하는 시스템의 전송 데이터 레이트에 의해서 결정되며 대부분 주어지는 값이 된다. 그리고 상기 정의되는 부호율 $R_{max}=(k/n)$ 의 정수 M배가 될 수 있는 최소의 부호율 R_{min} 을 정한다. 물론 상기 최소 부호율은 임의로 결정할 수 있으나 통상은 1/6 혹은 1/7 이하의 부호율을 사용한다. 이는 터보 부호의 경우 $R=1/7$ 이하에서는 부호율 감소에 따른 코딩 게인(Coding Gain)이 포화되는 성향을 보이기 때문이다. 그리고, 수신단 복호기의

실제 부호율(R, 모 부호율)을 정한다. 상기 부호율(R)은 ($R > R_{min}$)이 관계를 만족하도록 설계한다.

<44> 그리고, 상기 최대 부호율(R_{max}) 및 최소 부호율(R_{min})을 구한 후, 403단계에서 상기 최대 부호율과 최대 부호율을 가지고 하기 <수학식 2>을 이용해 서브 부호의 개수(M)을 구한다.

<45> **【수학식 2】** $M = \lceil R_{max}/R_{min} \rceil$

<46> 상기 서브 부호 개수를 구한 후, 405단계에서 변수 m을 '1'로 초기화하고, 407단계에서 $C(=m \times k)$ 를 결정한다. 여기서, 상기 C는 각 서브 부호 매트릭스의 컬럼(column)의 크기를 나타낸다. 상기 C는 상기 결정된 최대 부호율에 근거하여 결정되어진다. 예를 들어, 상기 최대 부호율이 3/4 인 경우, 상기 C는 3,6,9....가 될 수 있으며, 상기 가능한 크기들 중 최소의 값으로 결정한다. 여기서, 상기 최대 부호율을 3/4로 가정하였기 때문에, C는 '3'으로 설정한다. 그리고, 상기 변수 m과 상기 최대 부호율의 부호심볼수(n)을 곱해 변수 N_s 를 정의한다. 상기 변수 N_s 는 각 서브 부호 매트릭스에서 선택되어질 심볼수를 나타낸다.

<47> 그리고, 409단계에서 상기 N_s 가 상기 C 보다 큰지를 검사한다. 즉, 매트릭스에서 선택할 심볼의 수가 상기 매트릭스의 컬럼 사이즈보다 큰지를 검사한다. 여기서, 크다는 것은 상기 선택할 심볼의 수가 정보심볼의 개수를 넘는 것을 의미하고, 작다는 것은 상기 선택할 심볼의 수가 정보심볼의 개수보다 작다는 것을 의미한다. 만일, 크지 않다면 413단계로 진행하여 상기 N_s 가 2보다 큰지를 검사한다. 즉, 상

기 선택할 심볼의 수가 적어도 2개 이상인지를 검사한다. 만일, 2 이상이면, 415 단계로 진행하여 하기 수학식 3에 의해 turbo 부호기의 제 1부호기의 출력 제1부분과 turbo 부호기의 제2 부호기의 출력 제2부분에서 각각 심볼들을 선택한다.(혹은 천공하지 않는다) 하기 수학식 3에서 parity라 함은 ENC의 parity 심볼 부분을 선택한다는 의미이다.

<48>
$$\lfloor N_s/2 \rfloor \text{ ENC1}(\text{sys}+\text{parity})$$

【수학식 3】
$$\lfloor N_s/2 \rfloor \text{ ENC2}(\text{parity})$$

<49> 상기 '2'보다 작으면 417단계로 진행하여 상기 변수 m을 1만큼 증가시킨 후, 상기 407단계로 되돌아가 이하 과정을 재수행한다.

<50> 한편, 상기 변수 N_s 가 상기 C보다 크면, 411단계로 진행하여 상기 N_s 에서 상기 C를 감산한 값이 2 이상인지를 검사한다. 이는 정보심볼을 모두 선택하고도 적어도 2개의 선택할 심볼들이 존재하는지 검사하는 것이다. 이렇게 하는 이유는 터보부호는 기존의 다른 단일부호와는 도면 2에서 보듯이 두 개의 부속부호(Component code) ENC1과 ENC2가 병렬로 터보인터리버에 의해서 연결된 구조의 부호이다. 따라서 상기 N_s 에서 상기 C를 감산한 값이 2보다 작으면 결국 1개의 심볼만이 사용할 수 있는 것이고 따라서 다음과 같이 두 가지 경우만이 가능하다. 첫째는 이 한 개의 심볼을 ENC1(부속부호 1)의 패리티부분에 할당하는 방법이고 두 번째는 ENC2(부속부호 2)의 패리티 부분에 할당하는 방법이다. 그러나 두 경우 모두 터보부호의 관점에서 큰 문제점을 지니고 있다. 첫 번째의 경우에는 ENC2(

부속부호 2)에 할당되는 패리티 심볼이 전혀 없다. 따라서 결국 이 부호는 터보 부호가 아닌 ENC1(부속부호 1)만으로 구성된 K=4인 길쌈부호(convolutional codes)가 되고 터보부호가 가지는 터보인터리버 이득을 전혀 제공하지 못하는 결과를 초래한다. 두 번째의 경우에도 ENC1(부속부호 1)에서 보면 정보어심볼(systematic symbol)만이 전송되고 패리티 심볼이 전혀 사용되지 않으므로 부호율이 1인 부호가 되며 부호율이 1이므로 부호이득(coding gain)이 전혀 없는 부호화하지 않은 uncoded system 방식이 된다. 따라서 최소한 상기 N_s 에서 상기 C를 감산한 값이 2보다 작으면 터보부호의 성능을 제공할 수 없으며, 최소한 2 이상이 되어야 터보부호의 성능을 제공한다.

<51> 만일, 상기 411 단계 조건을 만족하면, 419단계로 진행하여 상기 C개의 정보심볼들을 선택한 후 소정 정해진 타입에 따라 나머지 심볼을 선택한다. 타입1인 경우, 421단계로 진행하여 하기 수학적식 4에 의해 제1패리티 부분과 제2패리티 부분의 심볼들을 각각 선택한 후 종료한다. 하기 수학적식 4에서 parity라 함은 ENC의 parity 심볼 부분을 선택한다는 의미이다.

<52>
$$\lceil (N_s - C)/2 \rceil \text{ ENC1}(\text{parity})$$

【수학적식 4】
$$\lfloor (N_s - C)/2 \rfloor \text{ ENC2}(\text{parity})$$

<53> 한편, 타입2인 경우 423단계로 진행하여 하기 수학적식 5에 의해 제1패리티 부분과 제2패리티 부분의 심볼들을 각각 선택한 후 종료한다. 상기 타입2는 미리 정해진 비율에 따라 제1패리티 부분과 제2패리티 부분의 심볼들을 선택하는 것을 의미한다. 하기 수학적식 5에서 parity라 함은 ENC의 parity 심볼 부분을 선택한다는 의미이다.

$$\begin{aligned} <54> \quad [a(N_s-C)/(a+b)] \text{ ENC1}(\text{parity}) \\ & \quad [b(N_s-C)/(a+b)] \text{ ENC2}(\text{parity}) \end{aligned}$$

【수학식 5】 $a+b=1$, 여기서 a, b 는 ENC1 및 ENC2의 심볼선택분배비를 나타냄.

<55> 한편, 411단계의 조건을 만족하지 않으면, 미리 정해진 타입에 따라 심볼들을 선택한다. 타입1인 경우 425단계로 진행하여 하기 수학식 6에 의해 제1부분과 제2부분의 심볼들을 각각 선택한 후 종료한다. 여기서, 제1부분의 심볼을 선택할 때 적어도 하나는 제1패리티부분에서 선택하도록 한다. 하기 수학식 6에서 parity라는 함은 ENC의 parity 심볼 부분을 선택한다는 의미이다.

$$\begin{aligned} <56> \quad [N_s/2] \text{ ENC1}(\text{단, 적어도 하나는 parity에서 선택}) \\ & \quad \text{【수학식 6】} \quad [N_s/2] \text{ ENC2}(\text{parity}) \end{aligned}$$

<57> 한편, 타입2인 경우 427단계로 진행하여 상기한 423단계와 동일하게 심볼들을 선택한 후 종료한다.

<58> 이하 C_1 부터 C_{s-2} 까지의 서브 부호 매트릭스를 구성하는 방법을 설명한다.. 여기서, C_1 에서부터 C_{s-2} 까지의 서브 부호 매트릭스들을 모두 생성할때 까지 상기 도 5의 알고리즘을 반복한다.

<59> 상기 도 8을 참조하면, 501단계에서 이전 구성된 매트릭스들을 참조하여 정보심볼들이 모두 선택되었는지를 검사한다. 만일, 상기 정보심볼들이 모두 선택되었다면 바로 507단계 혹은 509단계로 진행하며, 상기 정보심볼들이 모두 선택되지 않았다면 503단계로 진행하여 선택되지 않은 정보심볼들을 선택한다. 여기서, 상기 선택되는 정보심볼의 수를 N_{s1} 이라 정의한다. 이후, 505단계에서 상기 N_s 에서

상기 N_{s1} 을 감산하여 새로운 N_s 를 정의한다. 여기서, 새로이 정의된 N_s 는 각 서브부호 매트릭스에서 선택할 패리티 심볼수를 나타낸다.

<60> 상기 505단계에서 구해진 선택할 패리티 심볼수를 구한 후, 혹은 상기 501단계의 조건이 만족되면, 미리 정해진 타입에 따라 패리티 심볼들을 선택하게 된다. 만일 타입1인 경우, 507단계로 진행하여 하기 수학식 7에 의해 제1패리티부분과 제2패리티부분에서 각각 심볼들을 선택한다. 상기 507단계 및 상기 507단계를 수행함에 있어, 기존에 선택되지 않은 심볼들을 선택한다.

<61>
$$\lfloor N_s/2 \rfloor \text{ ENC1}(\text{parity})$$

【수학식 7】
$$\lfloor N_s/2 \rfloor \text{ ENC2}(\text{parity})$$

<62> 한편, 타입2인 경우 509단계로 진행하여 하기 수학식 8에 의해 제1패리티부분과 제2패리티부분에서 각각 심볼들을 선택한다.

<63>
$$\lfloor a(N_s)/(a+b) \rfloor \text{ ENC1}(\text{parity})$$

【수학식 8】
$$\lfloor b(N_s)/(a+b) \rfloor \text{ ENC2}(\text{parity})$$

<64> 이하 마지막 서브 부호 매트릭스(C_{s-1})를 구성하는 방법을 설명한다.

<65> 상기 도 9를 참조하면, 601단계에서 이전 구성된 매트릭스들을 참조하여 선택되지 않았던 심볼들을 선택한다. 여기서, 상기 선택된 심볼의 개수를 N_{s2} 라 정의한다. 그리고, 603단계에서 상기 N_s 에서 상기 N_{s2} 를 감산하여 새로운 N_s 를 정의한다. 여기서, 상기 새로이 정의된 N_s 는 반복할 심볼의 개수를 나타낸다. 그리고 605단계에서 상기 새로이 정의된 N_s 가 '0'보다 큰지를 검사한다. 만일, 동일하다

면 종료하고, 크다면 상기 N_s 만큼의 심볼의 개수를 반복한다. 여기서 상기 반복되는 심볼들은 가능한 정보심볼(systematic)이 반복되도록 한다.

<66> 예를 들어서, 시스템이 제공하는 가장 최대의 부호율 $R_{max}=3/4$ 이고, 시스템이 제공할 수 있는 복호기의 부호율 R 이 $1/5$ 인 경우 다음과 같이 R_{min} 과 M 이 결정된다.

<67> $R_{min}=1/6$: 최대 재전송의 한계에 의해서 결정된다.

<68> $M=6/(4/3)=4.5 \rightarrow 5$

<69> 따라서 Sub code set size: 5

<70> $\{C_0, C_1, C_2, C_3, C_4\}$: $R_{max}=3/4$

<71> Sub codes를 결합 $\rightarrow (1/M) \times R_{max} = (1/5) \times (3/4) = 3/20$

<72> $M \times b = 5 \times 4 = 20$, $5 \times a = 5 \times 3 = 15$, 따라서 $20 - 15 = 5$ 심볼을 정보심볼에서 반복되도록 한다.

<73> 전체 Sub codes를 결합하면 $R_{min}=3/20$ 이 된다. 따라서 $R=1/5$ 의 복호기를 사용하는 경우에는 20개의 부호심볼 중에서 5개의 심볼을 반복하면 된다. 따라서 각각의 서브 부호에서 정보심볼(Systematic symbol)인 X 를 1번씩 반복하도록 서브 부호를 구성하면, 복호기는 M 개의 서브 부호들을 모두 수신하는 경우 정보심볼이 2번 반복된 $R=1/5$ 터보 부호를 구할 수 있으며 이를 가지고 복호를 수행한다.

<74> 3. Redundancy Selection

<75> 3.1 Quasi-Complementary Code Set

<76> 패킷 부호 결합(Packet code combining)을 사용하는 대표적인 시스템에는 Hybrid ARQ 시스템이 있다. 즉, HARQ Type I, Type II, Type III에서 모두 패킷 부호 결합(Packet Code Combining)을 사용할 수 있으며, 이러한 시스템에 앞서 설명한 기술을 적용하고자 본 발명에서는 준 보완 터보코드 셋(Quasi-complementary turbo code set)을 사용한다.

<77> 예를 들어 패킷 전송을 위한 기본 단위의 정보비트 블록을 TU(transport unit)라고 정의하면 패킷 부호 결합(Packet code combining)을 사용하는 상기의 Hybrid ARQ 시스템 즉, HARQ Type I, Type II, Type III에서는 각각의 TU 전송 시마다 아래에 제시하는 준 보완 터보부호 셋(Quasi-complementary turbo code set) 중 하나의 서브 부호(sub code) C_i 가 할당된다.

<78> 물론 재전송의 기본단위는 초기 전송에 사용되는 TU의 크기와 동일한 것을 사용할 수도 있으며, 혹은 매 전송마다 다른 것을 사용할 수도 있다. 단, 한번의 초기 전송과 각각의 재전송 시마다 아래에 주어지는 준 보완 터보부호 셋(Quasi-complementary turbo code set)을 사용한다.

<79> 준 보완 터보부호(Quasi-complementary code) C_q 는 부호 셋 크기(code set size) S 를 가지며, 각각의 서브 부호(sub codes) C_i (여기서, $i=0,1,2,\dots,S-1$)를 조합함으로써 모 부호(mother code) C 를 복원(reconstruction)할 수 있거나 혹은 이 보다 작은 부호율을 가지는 새로운 C_q 를 구성(construction)할 수 있는 부호들을 의미한다. 여기서 모 부호라 함은 복호기(encoder)가 가질 수 있는 최소의 부호율을 의미한다. 이러한 모 부호를 사용하는 시스템에서 아래의 성질을 만족하는 경우 준 보완 터보부호(Quasi-complementary turbo codes)라 한다.

<80> Original code C with code rate $R=R_m$ or a code C with code rate $R<R_m$

<81>
$$= \bigcup_{i=0}^{S-1} C_i$$

<82> (where S is number of sub codes with code rate of R_i)

<83> 아래에 준 보완 터보부호(quasi-complementary turbo codes)를 사용하고, 전송 단위를 TU로 정의하고 초기전송과 재전송에서 모두 동일한 TU 크기를 사용하는 시스템에서 구체적인 전송방식을 보였다. 물론 각각의 전송에서 상이한 TU 크기를 사용하는 시스템에서의 전송방식도 제안이 가능하다. 하지만, 후술되어질 설명의 편의를 위해서 모두 동일한 TU 크기를 사용하는 시스템에서의 구체적인 전송방식을 보였다. 아래의 예는 서브 부호들(sub codes)의 수가 $S=4$ 이고, 모 부호율(mother code rate)이 $R=1/5$ 인 경우를 예를 보인 것이다.

<84> -초기전송 및 재전송 단위는 TU 단위로 이루어지며, 각각의 초기 전송 및 재 전송 시에 준 보완 부호 셋(quasi-complementary code set)의 서브 부호 C_i 가 각각 전송된다.

<85> -전체 부호율(Overall code rate)이 $1/5$ 보다 큰 경우에는 재전송이 요청될 때마다 준 보완 부호 셋(quasi-complementary code set)의 서브 부호 C_i 가 각각 전송되며, 그 순서는 $C_0, C_1, C_2, \dots, C_{S-1}$ 가 된다. 이 과정은 패킷 부호 결합(Packet code combining)을 수행하는 단계를 의미한다.

<86> -재전송과 초기전송을 포함한 수신된 패킷의 모든 연성결합에 의한 전체 부호율(Overall code rate)이 $1/5$ 보다 작거나 같은 경우에는 재전송이 요청될 때마

다 준 보완 부호 셋(quasi-complementary code set)의 서브 부호 C_i 가 각각 다시 반복되어 전송되며, 그 순서는 $C_0, C_1, C_2, \dots, C_{S-1}$ 이 된다. 이 과정은 패킷 다이버시티 결합(Packet diversity combining)을 수행하는 단계를 의미한다.

<87> - 준 보완 부호 셋 크기(Quasi-complementary code set size)는 임의의 값을 사용할 수 있으며, 이는 초기에 결정되는 복호기의 모 부호율에 의해서 결정된다. 여기서는 모 부호율 $R=1/5$ 이고, $R=2/3$ 인 부호율을 재전송을 위한 서브 부호율로 사용하므로 최대 4개의 서브 부호가 생성 가능하다. 따라서 $S=4$ 를 사용한다.

<88> 하기의 <표 3>에서는 준 보완 터보부호 셋(Quasi-complementary turbo codes sets)과 이를 사용하는 순방향 트래픽 채널에서의 패킷 전송율(Forward Traffic Channel packet data rates)의 일 예를 보이고 있다.

<89> 【표 3】

Set Size S	Code Set	Subcode Rate Set	Data Rates
1	$\{C_0\}$	$C_0: R_0=1/6$	307.2kbps 153.6kbps 76.8kbps 38.4kbps 19.2kbps
2	$\{C_0, C_1\}$	$C_0: R_0=1/3$ $C_1: R_1=1/3$	1228.8kbps 921.6kbps 614.4kbps 307.2kbps
4	$\{C_0, C_1, C_2, C_3\}$	$C_0: R_0=2/3$ $C_1: R_1=2/3$ $C_2: R_2=2/3$ $C_3: R_3=2/3$	2457.6kbps 1843.2kbps 1228.8kbps

<90> 상기의 <표 3>은 현재 IS-2000의 1XEV/DV에서 사용될 것으로 보이는 가용한 순방향채널의 전송율에 따라서 구한 것으로 모 부호율 $R=1/5$ 이고, 각각의 서브 부호의 부호율로 $R=2/3$, $1/3$, $1/6$ 을 사용하는 경우의 예를 보인 것이다. 상기 <표 3>에서 보여지고 있듯이 서브 부호율(sub code rate)이 $R=1/6$ 의 경우에는 모 부호율(Mother code rate) $R=1/5$ 보다 작으므로 매 전송 시마다 동일한 부호가 사용되며, 이를 C_0 로 표시하였다. 반면에 서브 부호율(sub code rate)이 $R=1/3$ 의 경우에는 모 부호율(Mother code rate) $R=1/5$ 보다 크므로 매 전송 시마다 상이한 부호가 사용되며 이를 C_0 , C_1 로 표시하였다. 이 경우 셋 크기(Set Size)는 $S=2$ 가 된다. 상기 서브 부호율(Sub code rate)이 $R=2/3$ 의 경우에는 모 부호율(Mother code rate) $R=1/5$ 보다 크므로 매 전송 시마다 상이한 부호가 사용되며, 이를 C_0 , C_1 , C_2 , C_3 으로 표시하였다. 이 경우 셋 크기(Set Size)는 $S=4$ 가 된다. 결국 각각의 서브 부호들이 셋 크기(Set Size) S 만큼 전송되는 경우 수신기는 원래의 모 부호율(mother code rate) R 을 복원할 수 있으며, 복호기(encoder)가 제공하는 최대의 부호화 이득(coding gain)을 제공할 수 있다.

<91> 4. Puncturing Matrix for Quasi-complementary codes

<92> 하기의 <표 4>에서는 상기 <표 3>에서 사용하는 각각의 부호율에 따른 천공 매트릭스(puncturing matrix)의 일 예를 보여주고 있다.

<93>

【표 4】

Code rates	C ₀	C ₁	C ₂	C ₃
R=1/6	$\begin{bmatrix} X \\ Y_0 \\ Y_1 \\ Y'_0 \\ Y'_1 \end{bmatrix} = \begin{bmatrix} 2 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$	NA	NA	NA
R=1/3	$\begin{bmatrix} 1 \\ 1 \\ 0 \\ 1 \\ 0 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 1 \end{bmatrix}$	NA	NA
R=2/3	$\begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$	$\begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}$
R=2/3	$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$

<94> 상기의 <표 4>에서 보여지고 있듯이 R-1/5 터보부호를 모 부호로 사용하고, 각각의 전송마다 R-2/3 부호를 사용한다고 하면 4개의 정보 심볼들 (information symbols)에 대해서 20개의 코드 심볼들(code symbols)이 발생된다. 이 중에서 14개를 천공(puncturing)하면 R-2/3 터보부호들(turbo codes)이 된다. 그러면 패킷 다이버시티 결합(packet diversity combining)은 전술한 천공 매트릭스(puncturing matrix)에 의해 생성되는 {C₀}을 재전송이 요구될 때마다 반복해서 전송하고, 이를 연성 결합(soft combining)하는 것을 의미한다. 반면에 패킷 부호 결합(packet code combining)은 아래와 같이 재전송 요청이 있을 때마다 서로 다른 부호 심볼들(coded symbols)을 전송하고, {C₀, C₁, C₂, C₄} 모두 전송된 이후에는 앞서와 유사하게 패킷 다이버시티 결합(packet diversity

combining)을 사용한다. 따라서 패킷 부호 결합(packet code combining)을 사용하는 HARQ Type III의 경우에는 4번의 전송이 이루어진 이후에 R=1/5의 전 코드 심볼들(full coded symbols)을 모두 가지고 복호(decoding)가 가능해 진다.

<95> 전술한 <표 4>에서 X, Y₀, Y₁, Y'₀, Y'₁은 각각 아래와 같이 R=1/5 터보 부호화기(Turbo encoder)에서 출력되는 부호 심볼들을 의미한다.

<96> X: Systematic code symbol (Information symbols)

<97> Y₀: Redundancy symbol from the upper component encoder of Turbo encoder

<98> Y₁: Redundancy symbol from the lower component encoder of Turbo encoder

<99> Y'₀: Redundancy symbol from the upper component encoder of Turbo encoder

<100> Y'₁: Redundancy symbol from the lower component encoder of Turbo encoder

<101> 본 발명의 실시 예에 따른 구체적인 터보 부호화기(Turbo encoder)의 구조는 도 2에서 보여지고 있는 바와 같다.

<102> 상기 도 2를 참조하면, 제1구성부호기 201은 입력되는 정보비트열을 부호화하여 부호심볼 X, Y₀, Y₁을 출력한다. 인터리버 202는 상기 입력되는 정보비트열을

주어진 규칙에 의해 인터리빙하여 출력한다. 제2구성부호기 203은 상기 인터리버 202로부터 출력되는 상기 인터리빙된 정보비트열을 부호화하여 부호심볼 X' , Y'_0 , Y'_1 을 출력한다. 심볼 천공/반복기 204는 상기 제1 및 상기 제2구성부호기 201, 203으로부터의 출력 부호심볼들 X , Y_0 , Y_1 , X' , Y'_0 , Y'_1 을 입력하고, 제어부 205로부터의 제어에 의해 상기 부호심볼들을 천공 및 반복하여 출력한다.

<103> 상기 도 2에서의 X' 는 실제로 전송되지 않는다. 따라서, 부호율은 $R=1/5$ 이 된다. 전술한 <표 4>에서 '1'은 전송되는 심볼을 의미하며, '0'은 전송되지 않고 천공(puncturing)되는 심볼을 의미한다. 또한 '2'는 해당되는 심볼이 2번 반복되어 전송되는 것을 의미한다.

<104> 상기의 <표 4>에서 $R=1/6$ 의 경우 전송되는 부호 심볼의 수열은 $X, X, Y_0, Y_1, Y'_0, Y'_1, X, X, Y_0, Y_1, Y'_0, Y'_1, \dots$ 와 같다. 따라서 실제 전송되는 부호 심볼의 수는 하나의 정보어 심볼(information symbol)에 대하여 6개의 부호어 심볼이 생성되므로 $R=1/6$ 이 된다. 상기 $R=1/6$ 의 천공 및 반복 매트릭스(Puncturing and Repetition Matrix)를 보면 복호기에서는 2번 반복된 X 심볼을 심볼연성결합(soft symbol combining)하여 하나의 심볼로 전환한 뒤에 복호를 수행한다. 따라서, 실제 복호기에 사용되는 부호율은 $R=1/5$ 로 간주된다. 이러한 정보어 심볼의 에너지가 증가된 $R=1/5$ 부호를 사용하는 경우에는 통상의 균일한 심볼 에너지를 사용하는 $R=1/5$ 부호에 비하여 성능이 개선된다. 또한 이러한 부호어 심볼을 반복에서 가장 적절한 위치는 바로 정보어 심볼이 된다. 이러한 관점에서 보면

$R=1/6$ 의 천공 및 반복 매트릭스(Puncturing and repetition matrix)를 사용함으로써 균일한 심볼 반복을 사용하고, 정보어 심볼을 반복하여 정보어 심볼의 에너지를 증가시키는 구조의 $R=1/6$ 부호를 사용할 수 있다.

<105> 상기한 <표 4>에서 $R=1/3$ 경우 전송되는 부호 심볼의 수열은 'X, Y0, Y'0, X, Y1, Y'1, '와 같다. 따라서 실제 전송되는 부호 심볼의 수는 하나의 정보어 심볼(information symbol)에 대하여 3의 부호어 심볼이 생성되므로 $R=1/3$ 된다. 그러나 각각의 전송에서 사용되는 C0, C1에서 사용한 천공 매트릭스(Puncturing matrix)가 다르므로 전송되는 심볼이 서로 상이하다는 것을 알 수 있다. 또한 C0과 C1을 연성결합하면 정보어 심볼인 X는 2번 반복되어 전송되고, 나머지 Y0, Y1, Y'0, Y'1은 각각 한번씩 전송된다. 따라서 상기 $R=1/6$ 인 경우와 동일하게 $R=1/5$ 의 복호기를 사용하여 복호가 가능하고 위에서 언급한 성질을 모두 만족하므로 성능을 보장 할 수 있다.

<106> 상기한 <표 4>에서 $R=2/3$ 첫 번째 경우 전송되는 부호 심볼의 수열은 'Y0, X, Y'0, X, Y'0, Y0, Y1, X, Y'1, X, Y'1, Y1, '와 같다. 따라서 실제 전송되는 부호 심볼의 수는 두개의 정보어 심볼(information symbol)에 대하여 3의 부호어 심볼이 생성되므로 $R=2/3$ 된다. 그러나 각각의 전송에서 사용되는 C0, C1, C2, C3에서 사용한 천공 매트릭스(Puncturing matrix)가 다르므로 전송되는 심볼이 서로 상이하다는 것을 알 수 있다. 또한 C0, C1, C2, C3을 연성결합하면 정보어 심볼인 X는 2번 반복되어 전송되고, 나머지 Y0, Y1, Y'0, Y'1은 각각 한번씩 전송된다. 따라서 상기 $R=1/6$ 인 경우와 동일하게 $R=1/5$ 의 복호기를 사용하

여 복호가 가능하고 위에서 언급한 성질을 모두 만족하므로 성능을 보장 할 수 있다.

<107> 상기한 <표 4>에서 $R=2/3$ 두 번째 경우 전송되는 부호 심볼의 수열은 'X, Y0, X, X, Y'0, X, Y'0, Y0, Y0, Y'0, Y1, Y1, Y'1, Y'1, Y1, Y'1, X, Y'1, X, X, Y1, X,'와 같다. 따라서, 실제 전송되는 부호 심볼의 수는 4개의 정보어 심볼(information symbol)에 대하여 6개의 부호어 심볼이 생성되므로 $R=2/3$ 된다. 그러나 각각의 전송에서 사용되는 C0, C1, C2, C3에서 사용한 천공 매트릭스(Puncturing matrix)가 다르므로 전송되는 심볼이 서로 상이하다는 것을 알 수 있다. 또한 C0, C1, C2, C3을 연성결합하면 정보어 심볼인 X는 2번 반복되어 전송되고, 나머지 Y0, Y1, Y'0, Y'1은 각각 한번씩 전송된다. 따라서 상기 $R=1/6$ 인 경우와 동일하게 $R=1/5$ 의 복호기를 사용하여 복호가 가능하고 위에서 언급한 성질을 모두 만족하므로 성능을 보장 할 수 있다.

<108> 5. 전송 프로토콜

<109> 패킷이 전송되는 트래픽 채널에 HARQ III를 적용하는 방식에 있어 전송하고자 하는 각각의 패킷의 전송 프로토콜은 순방향 트래픽 채널과 역방향 트래픽 채널 모두에서 사용 가능하므로 여기서는 특별히 전달하고자 하는 의미의 혼돈 가능성이 없는 경우에 순방향 트래픽 채널과 역방향 트래픽 채널을 구별하지 않고 총칭하여 트래픽 채널로 명칭한다.

<110> 5.1 전송패킷의 길이 및 물리채널 대응관계

<111> 패킷이 전송되는 트래픽 채널에 HARQ III를 적용하는 방식에 있어 전송하고자 하는 각각의 패킷의 길이는 가변적으로 결정할 수 있다. 여기서는 전송하고자 하는 하나의 패킷을 물리계층 패킷(PLP: Physical Layer Packet)으로 정의하고, 하나의 PLP에는 복수개의 서브 패킷이 존재할 수 있다. 이들을 각각 전송 유닛(TU: Transport Unit)이라고 명칭 한다. 상기 각각의 TU는 가변적인 크기를 가질 수 있다. 따라서 PLP 또한 가변적인 길이를 가질 수 있다. 물론 하나의 PLP에 한 개의 TU가 전달될 수도 있다. 따라서 아래에 기술되는 HARQ Type III의 프로토콜의 설명에서는 이를 구별하지 않고 대표되는 두 가지의 경우를 가지고 설명한다. 즉, 전송하고자 하는 패킷의 길이는 TU, 2TU, 3TU, 4TU를 가질 수 있고, 각각의 경우에 있어 최소의 TU당 비트수인 TU=768인 경우와 TU=1536인 경우 두 가지를 가진다. 여기서 TU=768인 경우를 이제부터 숏 포맷(Short Format)이라고 명칭하며, TU=1536인 경우를 롱 포맷(Long Format)이라고 명칭 한다. 따라서 아래에서는 숏 포맷(Short Format)과 롱 포맷(Long Format)의 두 가지 경우로 크게 나누어서 구별하고, 각각의 경우에 1TU, 2TU, 3TU, 4TU에 해당하는 PLP 크기를 가지는 경우의 재전송 방식과 HARQ type III방식에 관해서 기술한다. 물론 하나의 PLP에 포함되는 TU의 최대 수는 가변적이며, 물리전송채널이 제공하는 전송율에 따라서 결정되는 값이며 여기서는 편의상 4를 사용한다. 한편, 하나의 PLP를 전송하는 물리채널의 전송단위는 물리전송채널의 최소 전송단위로 이루어진다. 여기서는 편의상 이를 슬롯(Slot)으로 정의하며, 이를 기준으로 하나의 PLP를 전송한다. 하나의 PLP의 전송을 위한 슬롯(Slot)의 수는 1에서 임의의 수까지 설정

가능하다. 하나의 슬롯(Slot)에 전송되는 데이터의 수는 가변적이며, 물리전송채널이 제공하는 전송율에 따라서 결정되는 값이다. 즉, 하나의 PLP에 대해서 해당되는 데이터 전송율을 감안하여 슬롯(Slot)의 수가 결정된다. 여기서는 그 중 한 가지 예로서 하나의 PLP가 최소 1개의 슬롯(Slot)으로부터 최대 32개의 슬롯들(Slots)에 해당되는 숏 포맷(Short Format)의 경우와 하나의 PLP가 최소 2개의 슬롯(Slot)으로부터 최대 64개의 슬롯들(Slots)에 해당되는 롱 포맷(Long Format)의 경우를 고려한다. 이러한 구분은 앞서 TU=768과 TU=1536을 구분하는 것과 동일하며, TU=768이 최대 16개의 슬롯(slot)에 전송된다는 전제에서 이를 숏 포맷(Short Format)이라고 정의한다. 한편, 이의 두 배가되는 TU=1536 그리고 32 슬롯(slot)을 사용하는 것을 롱 포맷(Long Format)이라고 정의한다. 물론 이러한 구분은 전송하는 패킷의 길이에 따른 구분으로서 기본적으로 HARQ Type III의 프로토콜 동작 여부와는 큰 관계가 없으나 길이에 따른 시스템의 이득율(Throughput)의 변화와는 밀접한 관계가 있으므로 이를 구별하여 설명한다.

<112> 5.2 전송 트래픽의 오류검출방식 및 재전송 방식

<113> 하나의 PLP에 있는 각각의 TU는 독립적인 오류검출 부호를 가지고 있다. 따라서 각각의 TU에 발생하는 오류를 검출하여 재전송을 요청할 수 있으며, 혹은 복수개의 TU로 구성되는 하나의 PLP에 하나의 오류검출부호를 사용하여 전체 TU에 발생하는 오류를 검출하여 재전송을 요청할 수 있다. 실제로 수신기가 전송한 패킷에 관한 오류 여부를 검출하고 이를 송신기에 전송하는 과정에서 초기전송 및 재전송

의 단위는 PLP를 기준으로 동작하지만, 하나의 PLP에 존재하는 서브 블록인 복수개의 각각의 TU의 오류발생 여부에 따라서 재 전송되는 PLP 내의 TU 구성은 다를 수 있다.

<114> 도 3에서는 하나의 PLP가 하나의 슬롯(Slot)을 사용하는 경우의 HARQ의 전송방식을 보여주고 있다. 상기 도 3에서 보듯이 하나의 PLP는 인접한 3개의 슬롯들(slots)과 상호 엇혀(Interlacing) 전송된다. 즉, 상기 도 3에서 보듯이 4개의 슬롯(slot)에 대해서 각각 상이한 패킷이 전송될 수 있으며, 각각에 대해서 독립적인 ACK/NACK 신호가 역방향 채널을 통해서 전달된다. 따라서 이러한 구조에서 각각의 PLP는 독립적인 Stop-and-Wait ARQ(SW-ARQ) 프로토콜로 동작할 수 있으며, 항상 일정한 역방향 제어신호가 수신기로부터 송신기로 전달된다. 이러한 구조를 'Modulo N HARQ'라고 정의하고, N에 따라서 엇히게 되는 슬롯(slots)의 수를 정의한다. 상기 도 3에서는 N=4인 경우를 보인 것이다. 이러한 구조에서 엇혀진 슬롯들(Interlaced slots)을 하나의 사용자가 사용할 수도 있으며, 이 경우 각각의 연속한 슬롯(Slot)을 PLP전송을 위한 슬롯(Slot)으로 사용 가능하다. 그러나 이 경우에는 사용자의 HARQ Type III의 프로토콜이 SR-ARQ(Selective Repeat-ARQ)로 동작하며, N=4개의 슬롯(Slot) 만큼 전송된 데이터를 저장할 메모리가 수신기의 물리채널에 요구된다. 따라서, 여기서는 이러한 메모리 요구가 불필요한 SW-ARQ를 기준으로 설명한다. 그러나 SR-ARQ를 사용하는 경우에도 동일하게 프로토콜 설명이 가능하다.

<115> 도 4는 하나의 PLP가 두 개의 슬롯(Slot)을 사용하는 경우의 HARQ의 전송방

식을 보여주고 있는 도면이다. 상기 도 4에서 보듯이 하나의 PLP는 인접한 3개의 슬롯들(slots)과 상호 Interlacing된 2개의 슬롯(Slot)에 걸쳐서 전송된다. 따라서 수신기는 2개의 슬롯들(Slots)을 수신해야 완벽한 하나의 PLP를 복원할 수 있다. 상기 도 4에서 보듯이 4개의 슬롯(slot)에 대해서 각각 상이한 패킷이 전송될 수 있으며, 각각에 대해서 독립적인 ACK/NACK 신호가 역방향 채널을 통해서 전달된다. 따라서 이러한 구조에서 각각의 PLP는 독립적인 Stop-and-Wait ARQ 프로토콜로 동작할 수 있으며, 항상 일정한 역방향 제어신호가 수신기로부터 송신기로 전달된다. 이러한 구조를 'Modulo N HARQ'라고 정의하고, N에 따라서 Interlacing되는 슬롯들(slots)의 수를 정의한다.

<116> 송신기는 만일 복수개의 ACK/NACK 식별자 비트들(indicator bits) 중에서도 어느 한 개라도 NACK로 검출되면 순방향 트래픽 채널(Forward Traffic Channel)을 통해서 하기의 <표 5>와 <표 6>의 quasi-complementary code sets을 사용하여 재전송이 요청된 트래픽 채널의 PLP를 전송한다. 복수개의 ACK/NACK 식별자 비트들(indicator bits)이 사용되는 경우에 발생할 수 있는 ACK/NACK의 조합은 매우 다양하며, 각각의 경우에 송신기가 재전송 PLP를 통하여 전송하는 TU의 종류는 여러 방식에 의해서 전송할 수 있다. 일반적인 원칙으로서 다음과 같은 조건을 가지고 구성할 수 있다.

<117> 조건 1. 복수개의 TU에 해당하는 각각의 ACK/NACK 비트 중에서 ACK로 수신된 TU들은 전송하지 않는다.

<118> 조건 2. 복수개의 TU에 해당하는 각각의 ACK/NACK 비트 중에서 NACK로 수신된 TU들은 우선해서 전송하며, 이 때 전송의 우선순위는 서비스의 우선순위, 즉 각각의 TU에 따른 QoS에 우선적으로 전송한다.

<119> 조건 3. 재 전송하고자 하는 PLP에 주어진 슬롯 전체에 사용 가능한 총 비트수가 NACK로 수신된 TU에 해당하는 비트 수보다 큰 경우에는 상기 조건 2의 우선순위를 가지는 TU를 우선적으로 반복해서 전송한다.

<120> 조건 4. 단, 각각의 TU에 대한 QoS를 보전해야하는 경우에는 재 전송시에도 초기전송과 같은 QoS를 유지해서 전송하도록 각각의 TU에 대한 가중치를 계산해서 전송한다. 예로서 4개의 TU를 사용하는 PLP의 경우에 각각의 TU0, TU1, TU2, TU3에 할당된 QoS를 각각 QoS0, QoS1, QoS2, QoS3라하고 $(QoS0+QoS1+QoS2+QoS3)=1.0$ 이라고 가정하면 역방향 ACK 채널로부터 TU0가 ACK, 나머지 TU1, TU2, TU3이 NACK로 수신된 경우,

<121> $QoS1=QoS1*(1/(QoS0+QoS1+QoS2))$, 여기서 $(QoS0+QoS1+QoS2)<1.0$

<122> $QoS2=QoS2*(1/(QoS0+QoS1+QoS2))$, 여기서 $(QoS0+QoS1+QoS2)<1.0$

<123> $QoS3=QoS3*(1/(QoS0+QoS1+QoS2))$, 여기서 $(QoS0+QoS1+QoS2)<1.0$

<124> 로 사용하면 된다. 따라서 이를 일반화하면 다음과 같다. 하나의 PLP에 P개의 TU를 사용하고 각각을 TU0, TU1, TU2, ..., TU(P-1)라고 정의하면 이 중에서 NACK로 수신된 TU를 각각 TU(i), TU(j), ..., TU(s), $i, j, \dots, s \in \{0, 1, 2, 3, 4, \dots, P-1\}$ 이라고 하면 다음의 방식에 의해서 재전송 해야하는 TU들의 가중치를 계산하여 각각의 전송 비트수를 조정하여 재 전송한다.

<125> $QoS(i) = QoS(i) * (1 / (QoS(i) + QoS(j) + \dots + QoS(s)))$, 여기서
 $(QoS(i) + QoS(j) + \dots + QoS(s)) < 1.0$

<126> $QoS(j) = QoS(j) * (1 / (QoS(i) + QoS(j) + \dots + QoS(s)))$, 여기서
 $(QoS(i) + QoS(j) + \dots + QoS(s)) < 1.0$

<127> :

<128> $QoS(s) = QoS(s) * (1 / (QoS(i) + QoS(j) + \dots + QoS(s)))$, 여기서
 $(QoS(i) + QoS(j) + \dots + QoS(s)) < 1.0$

<129> 다음으로 트래픽 채널(Traffic Channel)은 역방향 ACK 채널(the Reverse Link ACK Channel)로 전달되는 복수개의 ACK/NACK 식별자 비트들(indicator bits) 모두에서 ACK가 검출되는 경우에만 새로운 순방향 트래픽 채널의 PLP를 전송한다.

<130> 5.3 트래픽 채널의 전송부호선택 방식

<131> 트래픽 채널로 전송되는 PLP는 매 전송 시마다 Qusi-complementary code를 사용하여 부호화(encoding)된다. 즉, 트래픽 채널의 PLP 전송율(data rate)에 따라 하기의 <표 5>와 <표 6>으로부터 설정되는 셋 크기(set size)가 S인 보완 부호 셋(complementary code set)에서 초기 전송에는 C0이 사용되고, 이후에 역방향 ACK 채널로부터 NACK가 전달될 때마다 트래픽 채널(Traffic Channel)로 C1, C2, CS-1, C0, C1, 의 순서로 전송되는 부호(code)가 순환(circulating)되어 선택된다.

<132> 【표 5】

Data Rates	Slots/ Physical Layer Packet	Transport Unit/ Physical Layer Packet	Code rate	Repetition	ACK/NACK Period	ACK Indicator bits /Reverse ACK channel	Quasi complementary code set
19.2kbps	32	1	1/6	16	2× slots	1	{C ₀ }
38.4kbps	16	1	1/6	8	2× slots	1	{C ₀ }
76.6kbps	8	1	1/6	4	2× slots	1	{C ₀ }
153.6kbps	4	1	1/6	2	2× slots	1	{C ₀ }
307.2kbps	2	1	1/6	1	2× slots	1	{C ₀ }
614.4kbps	1	1	1/3	1	1× slots	1	{C ₀ , C ₁ }
307.2kbps	4	2	1/3	2	2× slots	2	{C ₀ , C ₁ }
614.4kbps	2	2	1/3	1	2× slots	2	{C ₀ , C ₁ }
1228.8kbps	1	2	2/3	1	1× slots	2	{C ₀ , C ₁ , C ₂ , C ₃ }
921.6kbps	2	3	1/3	1	2× slots	3	{C ₀ , C ₁ }
1843.2kbps mnnn	1	3	2/3	1	1× slots	3	{C ₀ , C ₁ , C ₂ , C ₃ }
1228.8kbps	2	4	1/3	1	2× slots	4	{C ₀ , C ₁ }
2457.6kbps	1	4	2/3	1	1× slots	4	{C ₀ , C ₁ , C ₂ , C ₃ }

<133>

【표 6】

Data Rates	Slots/ Physical Layer Packet	Transport Unit/ Physical Layer Packet	Code rate	Repetition	ACK/NACK Period	ACK Indicator bits /Reverse ACK channel	Quasi complementary code set
19.2kbps	32	1	1/6	16	2×8 slots	1	{C ₀ }
38.4kbps	16	1	1/6	8	2×8 slots	1	{C ₀ }
76.6kbps	8	1	1/6	4	2×8 slots	1	{C ₀ }
153.6kbps	4	1	1/6	2	2×8 slots	1	{C ₀ }
307.2kbps	2	1	1/6	1	2×8 slots	1	{C ₀ }
614.4kbps	1	1	1/3	1	1×8 slots	1	{C ₀ , C ₁ }
307.2kbps	4	2	1/3	2	2×8 slots	2	{C ₀ , C ₁ }
614.4kbps	2	2	1/3	1	2×8 slots	2	{C ₀ , C ₁ }
1228.8kbps	1	2	2/3	1	1×8 slots	2	{C ₀ , C ₁ , C ₂ , C ₃ }
921.6kbps	2	3	1/3	1	2×8 slots	3	{C ₀ , C ₁ }
1843.2kbps	1	3	2/3	1	1×8 slots	3	{C ₀ , C ₁ , C ₂ , C ₃ }
1228.8kbps	2	4	1/3	1	2×8 slots	4	{C ₀ , C ₁ }
2457.6kbps	1	4	2/3	1	1×8 slots	4	{C ₀ , C ₁ , C ₂ , C ₃ }

<134> 예를 들어서 역방향 ACK 채널로부터 계속해서 3번의 NACK가 전달되면 순방향 트래픽 채널(Forward Traffic Channel)로는 각각의 NACK에 대응해서 C1, C2, C3의 순서로 차례대로 서브 부호(sub code)가 전달된다. 만일, 이후에도 계속해서 2번의 NACK가 전달되면 다시 C0, C1을 전달하고, 이후 ACK가 전달되면 전송을 중지하고, 새로운 트래픽 채널(Traffic Channel)의 PLP를 전송한다. 각각의 재전송에서 사용되는 quasi-complementary code 종류에 관한 정보는 송신기가 매번

전송하지 않고, 송신기와 수신기 모두 전송율(Data rate)에 따라 결정된 셋 크기(set size) S에 따라서 0, 1, 2, ..., S-1, 0, 1, ..., 의 순서로 순환(circulating)된다.

<135> 6. 역방향 ACK 채널의 구조 및 전송방식

<136> 순방향 트래픽 채널(Forward Traffic Channel)로 전송되는 각각의 PLP에 대하여 수신기는 PLP의 오류발생 여부를 역방향 ACK 채널(Reverse Link ACK Channel)로 전송한다. 상기 역방향 ACK 채널(Reverse Link ACK Channel)로 전송되는 ACK/NACK 메시지는 하나의 PLP에 있는 각각의 TU마다 하나의 ACK/NACK 식별자 비트(indicator bit)를 전송하므로 최대 4개의 독립적인 TU가 순방향 채널에서 전송되는 경우 역방향 채널에 사용되는 ACK/NACK 채널에도 4개의 ACK/NACK 비트가 전달된다. 물론 4개 이상의 복수개의 TU가 사용되는 경우에는 이에 해당하는 개수만큼의 ACK/NACK 비트가 전송된다.

<137> 6.1. 역방향 ACK 채널로 전송되는 ACK/NACK 신호의 주기성

<138> 다음으로 역방향 ACK 채널로 전송되는 ACK/NACK 신호의 주기성에 관해서 기술한다. 본 발명에서 제안하는 HARQ 방식에서는 SR-ARQ나 SW-ARQ 프로토콜에 관계없이 항상 일정한 ACK/NACK 전송주기를 사용하며, 아래와 같이 항상 일정한 ACK/NACK 신호의 주기성은 각각의 트래픽 전송율에 의해서 결정된다. 또한 트래픽 채널에서 전송되는 각각의 서브 부호(Sub code)의 부호길이가 1 슬롯(slot)

또는 최대 2 슬롯들(slots) 이내로 제한되므로 최대 2개의 슬롯(slot)을 수신하면 하나의 PLP에 대응되는 서브 부호(Sub code)를 반드시 수신할 수 있다. 즉, 트래픽 전송에 사용되는 모든 부호는 반드시 1 슬롯(slot) 혹은 2 슬롯들(slots)에서 전송이 완료되도록 설계된다. 따라서 정확한 ACK/NACK의 전송은 최대 2 슬롯들(slots) 혹은 최소 1 슬롯(Slot)의 단위로 전송되며, 항상 일정한 간격을 가진다. 상기 도 3과 상기 도 4는 전송한 바와 함께 각각의 전송율에 따른 서브 부호(sub codes)를 보였다.

<139> 상기 역방향 ACK 채널(Reverse Link ACK Channel)로 전송되는 ACK/NACK 메시지는 전송한 <표 5>와 <표 6>에 주어진 것처럼 트래픽 채널(Traffic Channel)의 PLP가 사용하는 패킷 전송율(Packet data rates), 패킷 포맷(Packet Format, Short Format/Long Format), 그리고 섞여진 멀티-슬롯(Interlaced multi-slot)을 사용하는가에 따라 상기 <표 5>와 상기 <표 6>으로 구분된다. 상기 역방향 ACK 채널(Reverse Link ACK Channel)로 전송되는 ACK/NACK 메시지는 상기 트래픽 채널(Traffic Channel)의 PLP가 숏 포맷(Short Format)을 사용하고, 4 슬롯의 섞여진 멀티-슬롯(interlaced multi-slot) 구조를 사용하는 경우 각각의 PLP 전송율(data rates)에 따라서 주기가 4 슬롯(=5msec)인 것과 8 슬롯(=10msec)인 두 가지 경우로 구분된다. 즉, 역방향 ACK 채널로 ACK/NACK를 전송하기 위해서 수신기는 전송 트래픽을 수신한 시점으로부터 1 슬롯이 진행된 후에 두 번째 슬롯의 앞 쪽 반 슬롯에 시간 다중화(time multiplexing)되어 송신기로 전송된다. 따라서 수신기로부터 전송 트래픽을 수신한 시점으로부터 정확하게 1 슬롯이 진행된 후에 항상 ACK/NACK가 전송된다.

<140> 상기 트래픽 채널(Traffic Channel)의 PLP 전송 후에 처음으로 역방향 ACK 채널(Reverse ACK Channel)로부터 전송되는 ACK/NACK을 1번 ACK/NACK이라 정의할 때, 주기가 4 슬롯인 경우 상기 도 3과 같이 홀수/짝수 번째 구별 없이 각각 전송되는 ACK/NACK는 빠른 종료(Early termination)를 위해 사용된다. 또한 상기 순방향 트래픽 채널(Forward Traffic Channel)의 PLP의 재전송 시 준 보완 서브 부호(quasi-complementary sub code)의 전환을 위한 제어 메시지(control message)로도 사용된다. 상기 도 4와 같이 주기가 8 슬롯인 경우 홀수 번째와 짝수 번째 관계없이 전송되는 모든 ACK/NACK는 빠른 종료(Early termination)를 위해 사용되며, 짝수 번째 전송되는 ACK/NACK만이 순방향 트래픽 채널(Forward Traffic Channel)의 PLP의 재전송 시 준 보완 서브 부호(quasi-complementary sub code)의 전환을 위한 제어 메시지(control message)로 사용된다. 여기서 빠른 종료(Early termination)라 함은 하나의 PLP에 할당된 슬롯을 모두 수신하기 이전에 이보다 작은 제한된 슬롯을 수신한 상태에서 전송하고자 하는 PLP를 오류 없이 수신할 수 있는 경우 더 이상의 슬롯 전송을 트래픽 채널로 전송하지 않고 새로운 PLP를 전송함으로써 전송효율을 증가시키기 위한 방식을 말한다.

<141> 상기 트래픽 채널(Traffic Channel)의 PLP가 상기 롱 포맷(Long Format)을 사용하는 경우에는 PLP 전송율(data rates)에 따라서 주기가 8 슬롯(=10msec)인 것과 16 슬롯(=20msec)인 두 가지 경우로 구분되며, 홀수 번째와 짝수 번째의 ACK/NACK의 역할은 상기 숏 포맷(Short Format)의 경우와 동일하다.

<142> 6..2 전송프로토콜의 동작 예

- <143> 도 5와 도 6은 본 발명의 실시예에 따른 동작의 예를 보이고 있는 도면이다. 앞서 언급하였듯이 ACK/NACK에 의한 제어상태 변화는 1 슬롯/PLP와 2 슬롯/PLP의 두 가지로 구분되며, 모든 전송율에 대해 상기 두 가지 경우 중 어느 한가지가 사용된다.
- <144> 상기 도 5에는 2 슬롯/PLP의 경우에 HARQ 동작에 따른 슬롯 처리 흐름을 보여주고 있는 도면이다. 상기 도 6에는 1 슬롯/PLP의 경우에 HARQ 동작에 따른 슬롯 처리 흐름을 보여주고 있는 도면이다. 상기 두 방식 사이의 차이는 재전송 시에 사용하는 서브 부호(Sub code)의 전환을 2 슬롯단위로 수행하는가와 아니면 1 슬롯단위로 가능한가의 차이점이다.
- <145> 상기 도 10과 도 11 및 도 12에서 보여지고 있는 바와 같이 모든 경우에 있어서 ACK/NACK는 매 슬롯마다 전송되며, 이를 구분해서 두 가지로 사용한다.
- <146> 이하 상기 첨부된 도면을 참조하여 본 발명의 실시예에 따른 동작을 상세히 설명하면 다음과 같다. 우선 후술할 본 발명의 실시예에 따른 상세한 동작은 무선 통신시스템의 송신장치에서 수행되는 동작과 수신장치에서 수행되는 동작을 구분하여 설명할 것이다.
- <147> 먼저, 도 10에서 보여지고 있는 제어 흐름을 참조하여 송신장치에서의 동작을 상세히 설명하면, 송신장치는 710단계에서 전송할 PLP를 부호화하여 서브 부호 셋을 생성한다. 상기 서브 부호 셋을 생성하기 위한 부호화를 수행하는 부호화기의 일 예로 도 2에서 보여지고 있는 터보 부호화기를 들 수 있다. 한편, 상기 서브 부호 셋은 상기 서브 부호 셋 크기에 의해 개수가 결정되는 서로 다른 복수의 서브 부호들로 구성된다. 이와 같은 구성을 가지는 것은 상기 서브 부호

셋이 서로 다른 복수의 서브 부호들로 구성되는 것은 패킷 코드 컴바이닝을 적용하기 위함이다.

<148> 상기 서브부호 셋의 생성이 완료되면 상기 송신장치는 712단계로 진행한다.

상기 송신장치는 712단계로 진행하여 상기 복수의 서브부호들 중 전송되는 서브 부호를 카운트하기 위해 소정 카운트 값(i)을 0으로 세팅한다. 즉, 상기 카운트 값 i 는 상기 모든 서브부호들의 전송이 완료되었는지를 판단하기 위해 사용된다. 상기 i 를 0으로 설정한 후 상기 송신장치는 714단계로 진행한다. 상기 송신장치는 714단계에서 상기 서브부호들 중 i 번째 전송 프레임(C_i)을 전송한다. 상기 송신장치는 상기 i 번째 서브부호를 전송한 후 716단계로 진행한다. 상기 송신장치는 상기 716단계에서 상기 전송한 i 번째 서브부호에 응답하여 상기 수신장치로부터 ACK가 수신되는 지를 판단한다. 한편, 상기 714단계와 상기 716단계에서는 전송단위에 대응하여 ACK를 전송하는 경우를 설명하고 있으나 앞에서 밝힌 바와 같이 전송 프레임을 통해 전송되는 서브 부호에 대응하여 ACK를 전송하는 형태로 구현할 수 있음은 자명하다. 또한, 상기 도 3, 도 4를 참조하면, 하나의 전송단위(슬롯)을 통해 하나의 서브 부호를 전송하는 경우에는 서브 부호 각각에 대응하여 ACK 또는 NACK를 수신하게 된다. 상기 서브부호에 응답하여 ACK 또는 NACK를 전송하는 수신장치의 구체적인 동작은 도 8과 도 9를 참조하여 후술될 것이다.

<149> 상기 송신장치는 상기 716단계에서 ACK의 수신을 감지하면 상기 710단계로 리턴하여 다음 전송할 PLP을 부호화하여 전송하기 위한 일련의 동작을 수행한다. 이에 반하여 상기 716단계에서 ACK가 수신되지 않았다는 것은 NACK(재 전송

요구)를 수신하였다는 것을 의미한다. 따라서, 상기 송신장치는 상기 716단계에서 NACK가 수신되었음을 감지하면 718단계로 진행한다. 상기 송신장치는 상기 718단계에서 PLP 단위로의 전송이 완료되었는가를 판단한다. 상기 718단계에서 해당 PLP 단위로의 전송이 완료되었다고 판단하면 720단계로 진행하며, 전송이 완료되지 않았다고 판단되면 722단계로 진행한다. 상기 송신장치는 상기 720단계에서 다음으로 전송할 서브부호를 선택하기 위해 상기 I를 1 증가시킨 후 722단계로 진행한다. 이는, 상기 송신장치가 동일한 PLP에 대해 코드 전환을 하여 전송하도록 하기 위함이다.

<150> 상기 송신장치는 722단계에서 상기 I를 상기 서브부호들의 총 개수(S)와 비교하여 모든 서브부호들에 대한 전송이 완료되었는가를 판단한다. 만약, 상기 송신장치는 상기 722단계에서 I가 S보다 크지 않으면 아직 상기 모든 서브부호들에 대한 전송이 이루어지지 않았다고 판단하여 상기 714단계로 리턴한다. 상기 송신장치는 상기 714단계로 리턴하면 상기 I에 의해 선택된 다음 전송할 서브부호(C_i)를 전송한다. 이에 반하여, 상기 송신장치는 상기 722단계에서 I가 S보다 크면 상기 모든 서브부호들의 전송이 완료되었다고 판단하여 712단계로 리턴하여 전송한 동작에 의해 상기 서브부호들에 대한 재 전송 동작을 수행한다. 따라서, 상기 재 전송 동작에 의해서는 상기 최초 전송된 서브 부호들과 동일한 서브 부호들이 전송되어 패킷 다이버시티 컴바이닝이 수행되도록 하기 위함이다.

<151> 전송한 바와 같이 상기 서로 다른 복수의 서브 부호들은 수신장치로부터의 재 전송 요구(NACK)의 수신에 의해 순차적으로 전송하여야 함에 따라 상기 서로 다

른 복수의 서브 부호들 각각은 상기 PLP를 의미할 수 있다. 예를 들어, 상기 서브 부호 셋이 $C_0, C_1, C_2, \dots, C_{S-1}$ 의 서브 부호로 구성될 수 있다면 상기 송신장치는 최초에 C_0 을 상기 수신장치로 전송할 것이며, 상기 수신장치는 상기 C_0 가 정상적으로 수신되었다면 상기 PLP의 수신을 종료하고 다음 PLP를 수신하기 위한 동작을 수행한다. 하지만, 상기 수신장치가 상기 C_0 를 정상적으로 수신하지 못하였다면 상기 송신장치는 상기 수신장치로부터의 재 전송 요구(NACK)를 수신하여 상기 서브 부호 셋을 구성하는 다음 서브 부호인 C_1 을 전송한다. 이를 위해서는 상기 송신장치로부터 전송되는 각각의 서브 부호들 각각은 상기 PLP로서의 의미를 가진다고 할 수 있다.

<152> 다음으로, 도 11과 도 12에서 보여지고 있는 제어 흐름을 참조하여 송신장치에서의 동작을 상세히 설명하면, 수신장치는 810단계에서 다이버시티 컴바이닝의 수행 여부를 판단하기 위한 카운트 값(j)과 서브부호 셋을 구성하는 모든 서브부호들의 수신 여부를 판단하기 위한 카운트값(i)을 0으로 셋팅한다. 본 발명에서는 앞에서도 밝힌 바와 같이 부호 컴바이닝과 다이버시티 컴바이닝을 혼용하여 사용하고 있다. 즉, 최초 전송되는 PLP에 대해서는 부호 컴바이닝만을 사용하여 복호를 수행하며, 최초 전송되는 PLP 이후에 전송되는 PLP에 대해서는 다이버시티 컴바이닝을 우선적으로 수행한 후 부호 컴바이닝을 수행한다.

<153> 상기 송신장치는 812단계에서 송신장치로부터의 서브부호(C_i)를 수신한다. 상기 수신장치는 상기 서브부호를 수신하면 814단계에서 상기 수신한 서브부호가 상기 송신장치로부터 최초로 전송된 PLP에 속하는 지를 판단한다. 상기 판단은 상기 j 를 한 PLP를 구성하는 전송 프레임의 총 개수(S)와 비교함으로써 가능하다

. 즉, 상기 j 가 상기 S 보다 크지 않다면 상기 수신한 서브부호는 최초 PLP에 속한다고 할 것이다. 하지만, 상기 j 가 상기 S 보다 크다면 상기 수신한 서브부호는 최초 PLP에 속하지 않는다고 할 것이다. 상기 814단계에서 상기 j 가 상기 S 보다 크다고 판단하면 상기 송신장치는 816단계로 진행하며, 상기 j 가 상기 S 보다 크지 않다고 판단하면 상기 송신장치는 818단계로 진행한다.

<154> 상기 수신장치는 상기 816단계로 진행하면 상기 수신한 전송 프레임에 대해 다이버시티 컴바이닝을 수행한다. 상기 전송 프레임에 대해 수행하는 다이버시티 컴바이닝은 상기 수신한 서브 부호와 이전에 수신한 서브 부호들 중 상기 서브 부호에 대응하는 서브 부호간에 수행된다. 예컨대, 상기 이전 서브 부호들이 $C_0, C_1, C_2, \dots, C_{S-1}$ 이고, 상기 수신한 서브 부호가 C_0 이면 상기 816단계에서는 이전의 서브 부호 C_0 과 상기 이후에 수신한 C_0 간의 다이버시티 컴바이닝을 수행한다.

<155> 하지만, 상기 송신장치는 상기 814단계에서 j 가 S 보다 크지 않다고 판단되거나 상기 816단계에서 다이버시티 컴바이닝의 수행이 완료되면 818단계로 진행하여 코드 컴바이닝을 수행한다. 상기 818단계에서 수행하는 코드 컴바이닝의 상세한 동작은 도 9에서 제어 흐름으로 보여지고 있다.

<156> 상기 도 13를 참조하여 코드 컴바이닝의 상세 동작을 설명하면, 상기 송신장치는 910단계에서 상기 수신한 전송 프레임에 대한 코드 컴바이닝을 수행한다. 상기 코드 컴바이닝은 하기의 <수학식 9>에 의해 계산되어질 수 있다.

<157> **【수학식 9】** $C_0 \cup C_1 \cup C_2 \dots \cup C_i$

<158> 상기 <수학식 9>에서 알 수 있는 바와 같이 상기 코드 컴바이닝은 I번째 서브 부호들의 교집합에 의해 계산된다. 상기 코드 컴바이닝에 의해 부호어가 결정되면 상기 수신장치는 912단계로 진행하여 상기 결정된 부호어에 의해 수신한 서브부호의 정보에 대한 복호를 수행한다. 상기 수신장치는 914단계에서 상기 복호된 정보들에 대해 오류 검사를 수행한다. 상기 오류 검사는 상기 서브부호를 통해 제공받은 CRC를 통해 수행한다. 상기 오류 검사가 완료되면 상기 수신장치는 916단계에서 오류 발생 여부를 판단한다. 상기 판단에 의해 상기 수신한 전송 프레임에 오류가 발생하였다고 판단하면 상기 수신장치는 918단계로 진행하여 재 전송 요구에 해당하는 NACK를 전송한다. 하지만, 상기 판단에 의해 오류가 발생하지 않았다고 판단되면 상기 수신장치는 920단계로 진행하여 정상적인 전송이 이루어졌음을 알리는 ACK를 상기 송신장치로 전송한다.

<159> 한편, 상기 송신장치는 상기 918단계에서 NACK를 전송한 후 922단계로 진행하여 상기 수신한 C_i 를 저장한다. 상기 수신장치는 상기 920단계에서 ACK를 전송하였으면 상기 810단계로 진행한다. 하지만, 상기 918단계에서 NACK를 전송하였으면 상기 922단계에서 해당 서브 부호의 저장을 완료한 후 상기 도 8의 820단계로 진행한다. 상기 820단계로 진행한 상기 송신장치는 상기 j 를 1 증가시킨 후 다음으로 수신되는 서브부호의 수신을 위해 상기 812단계로 리턴한다.

<160> 전송한 바와 같이 본 발명의 실시 예에 따른 수신장치는 송신장치로부터의 서브부호에 대응하여 ACK 또는 NACK를 전송하며, 상기 ACK를 전송한 경우에는 송신장치로부터의 다음 PLP에 의한 서브부호를 수신한다. 하지만, 상기 NACK를 전송한 경우에는 송신장치로부터의 동일한 PLP에 의한 서브부호를 수신한다. 따라

서, 상기 수신장치는 정상적인 서브부호의 수신에 따른 ACK를 전송할 때까지 최대 S개의 서브부호들을 수신하여 코드 컴바이닝을 수행하며, 상기 S개의 서브부호에 의해서도 정상적인 수신이 이루어지지 않는다면 상기 수신하지 못한 PLP의 수신을 위한 동일한 수신 동작을 반복하여 재 수행하게 된다.

【발명의 효과】

<161> 상술한 바와 같이 본 발명은 패킷 부호 컴바이닝과 다이버시티 컴바이닝을 전송율에 따라 가변적으로 사용함으로써 보다 효율적인 데이터 전송을 할 수 있는 이점이 있다. 또한, 터보 코드를 사용하는 통신시스템에서 재전송방식을 사용하는 경우 시스템의 이득율을 개선하고자 연성 결합(Soft combining)을 사용하며, 특히 터보 코드의 특성을 고려한 패킷 코드 컴바이닝(packet code combining)을 사용하여 시스템의 이득율(throughput)을 개선하는 효과가 있다. 마지막으로, 본 발명은 재전송방식을 사용하는 패킷 통신시스템 혹은 재전송을 사용하는 일반적인 통신시스템에서 오류정정부호와 오류검출을 결합한 복합재전송 방식의 시스템 성능 향상시킬 수 있다.

【특허청구범위】**【청구항 1】**

무선 통신시스템의 송신장치에서 물리 계층 패킷을 소정 부호화 절차에 의해 서브 부호 셋을 생성하여 전송하는 방법에 있어서,

상기 서브 부호 셋을 소정 단위로 분할한 복수의 전송 프레임들을 순차적으로 전송하고, 상기 전송한 적어도 하나의 전송 프레임에 응답하여 재 전송 요구가 수신되는 지를 감시하는 과정과,

상기 재 전송 요구를 수신하면 상기 복수의 전송 프레임들의 전송을 완료한 후 상기 복수의 전송 프레임들을 순차적으로 재 전송하는 과정을 포함함을 특징으로 하는 상기 방법.

【청구항 2】

무선 통신시스템의 송신장치에서 물리 계층 패킷을 전송하는 방법에 있어서,

상기 물리 계층 패킷을 소정 부호화 절차에 의해 서브 부호 셋을 생성하고, 상기 서브 부호 셋을 구성하는 서로 다른 복수의 서브 부호들을 수신장치로부터의 재 전송 요구를 기준으로 하여 패킷 코드 컴바이닝방식에 의해 전송하는 과정과,

상기 패킷 부호 결합방식에 의한 상기 서로 다른 복수의 서브 부호들이 모두 전송될 때까지 상기 수신장치로부터의 ACK를 수신하지 못하면 상기 서로 다른

복수의 서브 부호들을 패킷 다이버시티 컴바이닝방식에 의해 전송하는 과정을 포함함을 특징으로 하는 상기 방법.

【청구항 3】

제2항에 있어서, 상기 패킷 코드 컴바이닝은,

상기 수신장치로부터의 ACK를 수신할 때까지 상기 재 전송 요구에 대응하여 상기 서로 다른 복수의 서브 부호들을 순차적으로 전송함을 특징으로 하는 상기 방법.

【청구항 4】

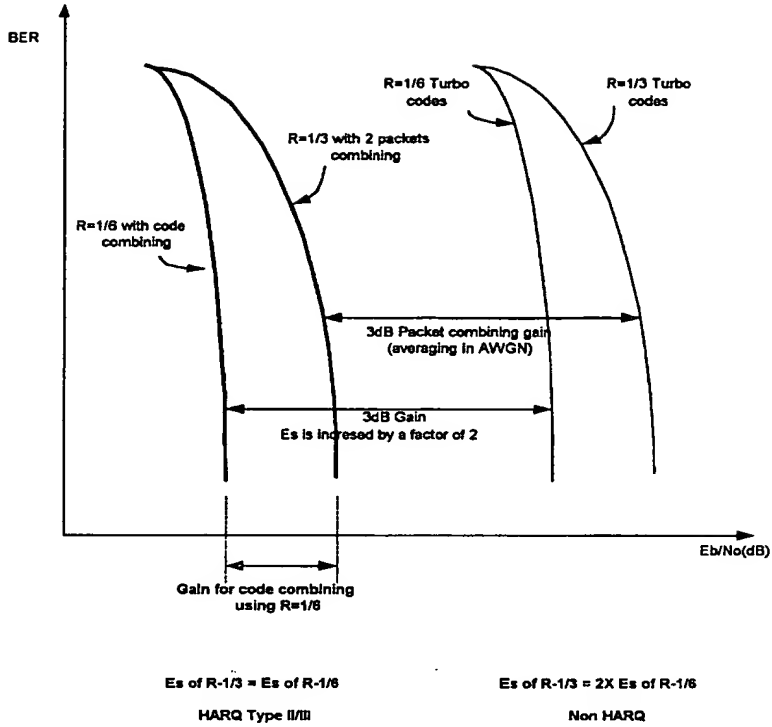
무선 통신시스템의 수신장치에서 송신장치로부터의 물리 계층 패킷을 수신하여 복호하는 방법에 있어서,

상기 송신장치로부터 상기 최초 물리 계층 패킷이 소정 부호화 절차에 의해 생성된 서브 부호 셋의 모든 서브부호들을 패킷 코드 컴바이닝을 통해 오류가 발생하지 않을 때까지 재 전송 요구하는 과정과,

상기 재 전송 요구에 의해 재 전송되는 상기 서브부호들을 패킷 다이버시티 컴바이닝과 상기 패킷 코드 컴바이닝을 수행하고, 상기 패킷 다이버시티 컴바이닝과 상기 패킷 코드 컴바이닝을 통해 오류가 발생하지 않을 때까지 재 전송을 요구하는 과정을 포함함을 특징으로 하는 상기 방법.

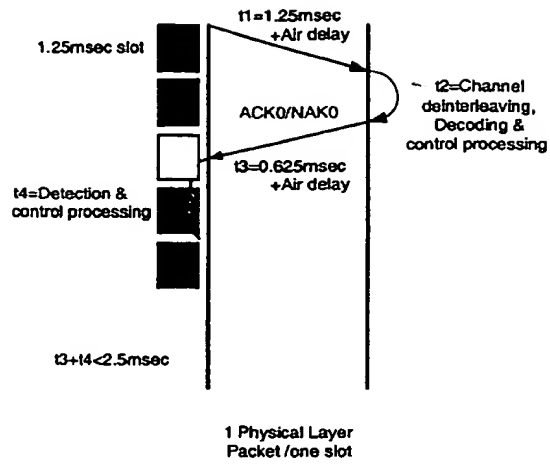
【도면】

【도 1】

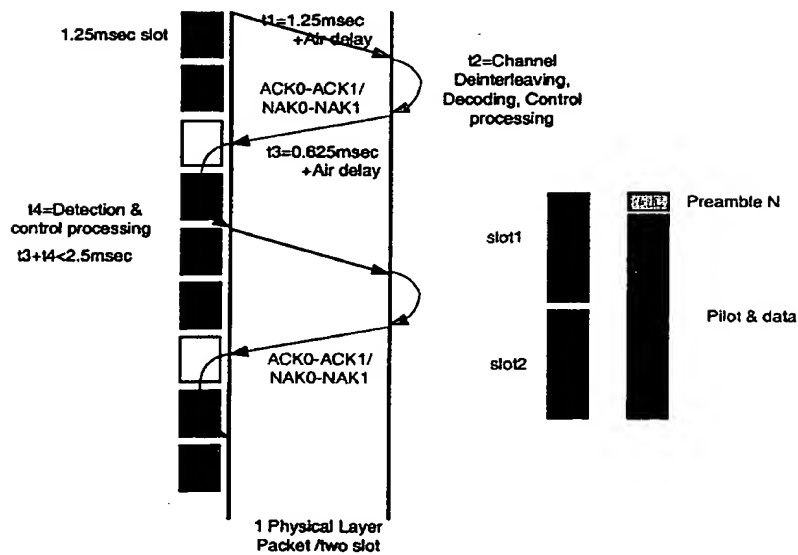


The diagram illustrates a turbo encoder system. It consists of two constituent encoders, labeled 201 and 203, which are identical in structure. Each encoder receives an input signal X and a control signal. The control signal is derived from a '터보 인터리버' (Turbo Interleaver) block, labeled 202. The output of the turbo encoder system is a set of signals Y_0 , which are fed into a '서브부호 발생기' (Sub-code Generator) block, labeled 204. A '제어기' (Controller) block, labeled 205, provides control signals to both constituent encoders and the sub-code generator. The internal structure of each constituent encoder includes a series of blocks and feedback loops, with labels n_0 , n_1 , and d indicating specific components or signals within the feedback paths.

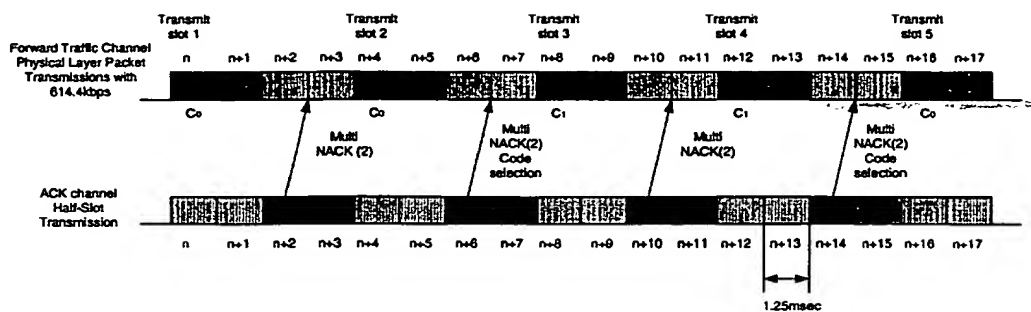
【도 3】



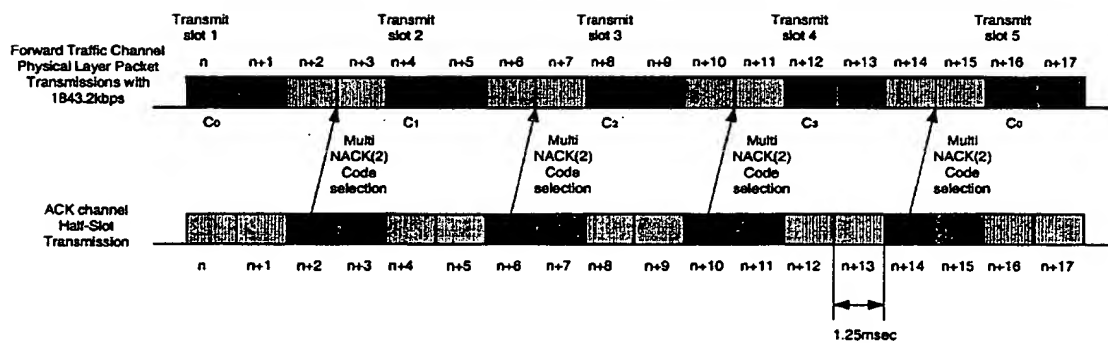
【도 4】



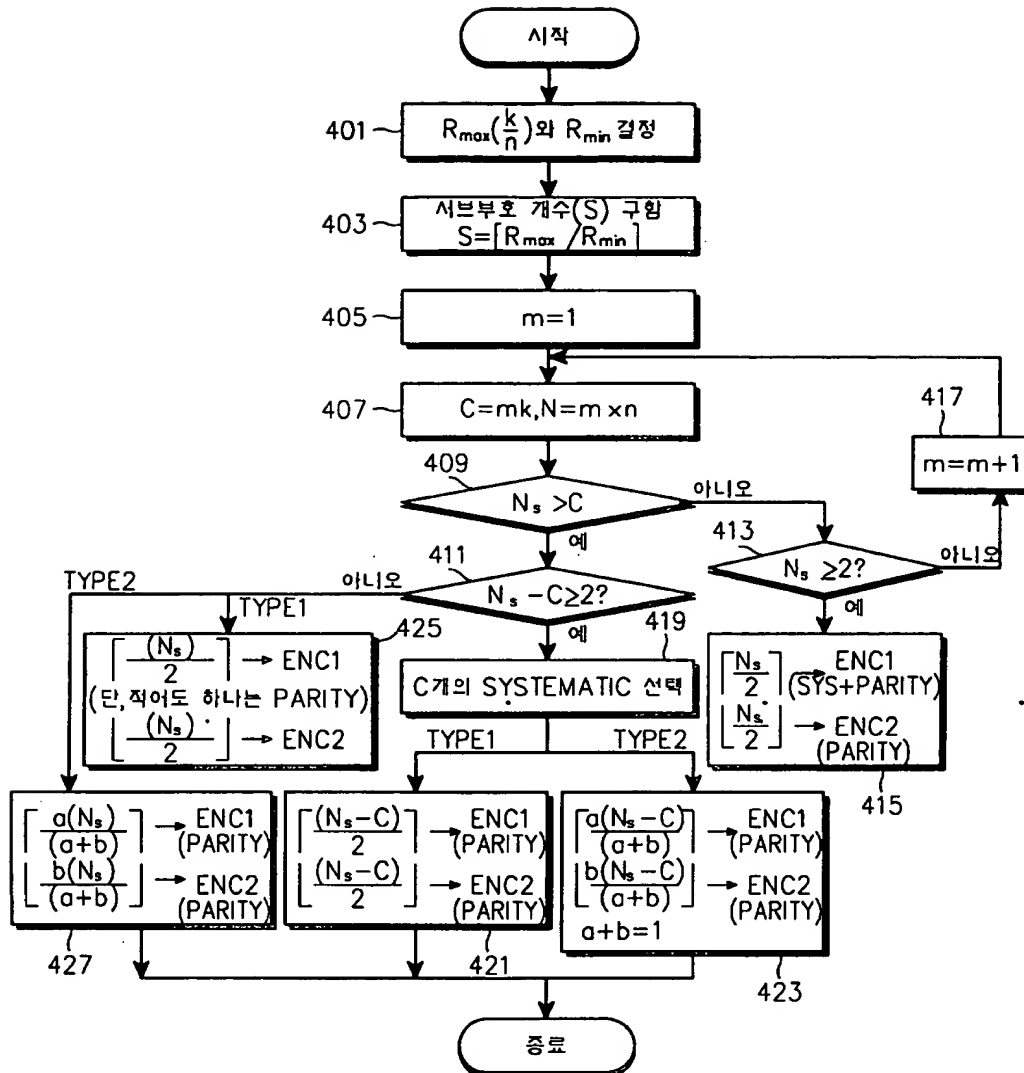
【도 5】



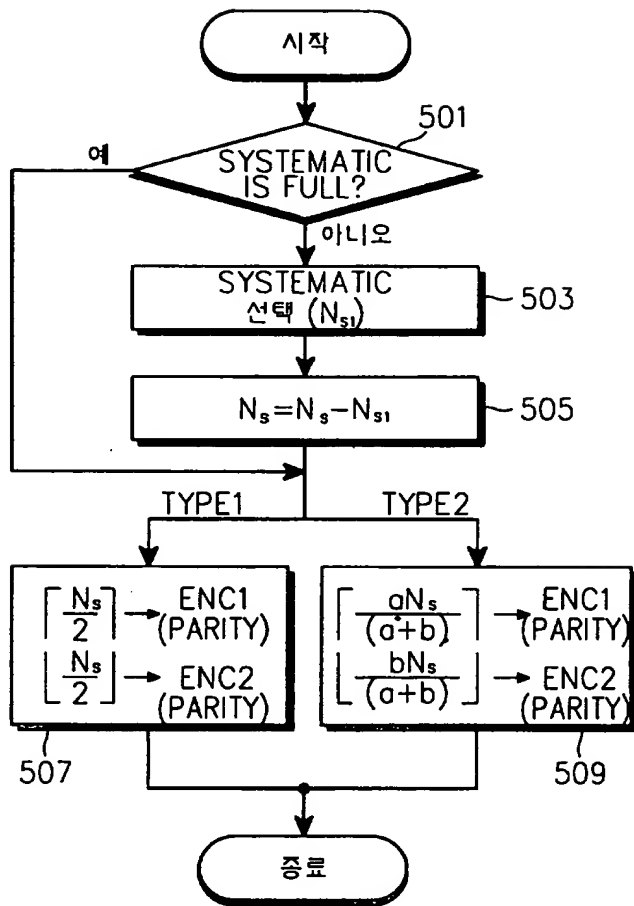
【도 6】



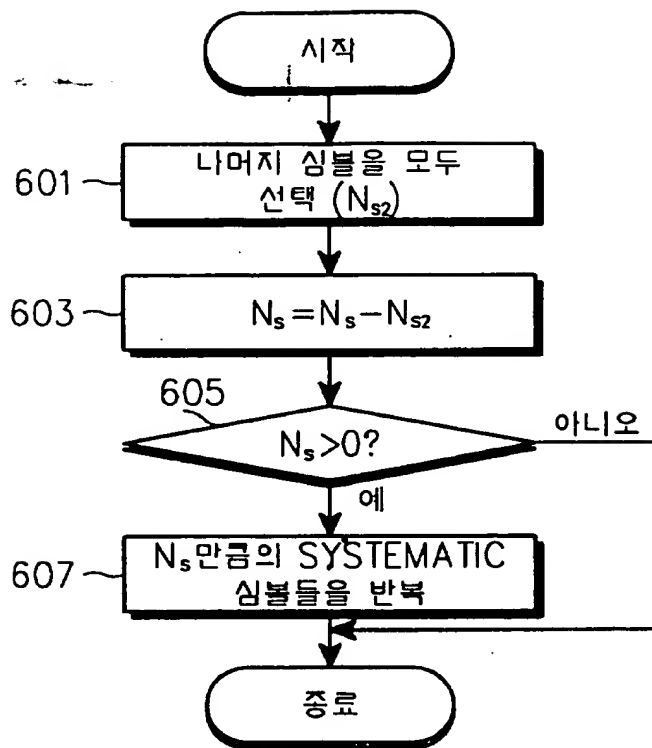
【도 7】



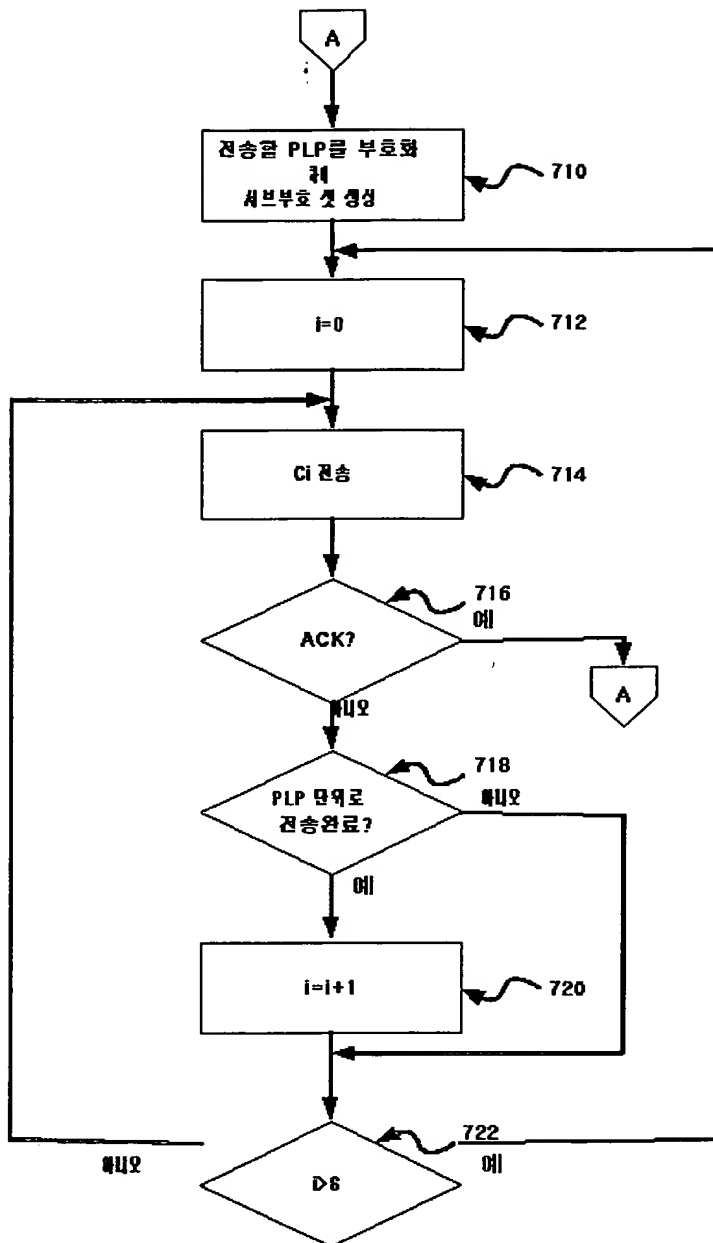
【도 8】



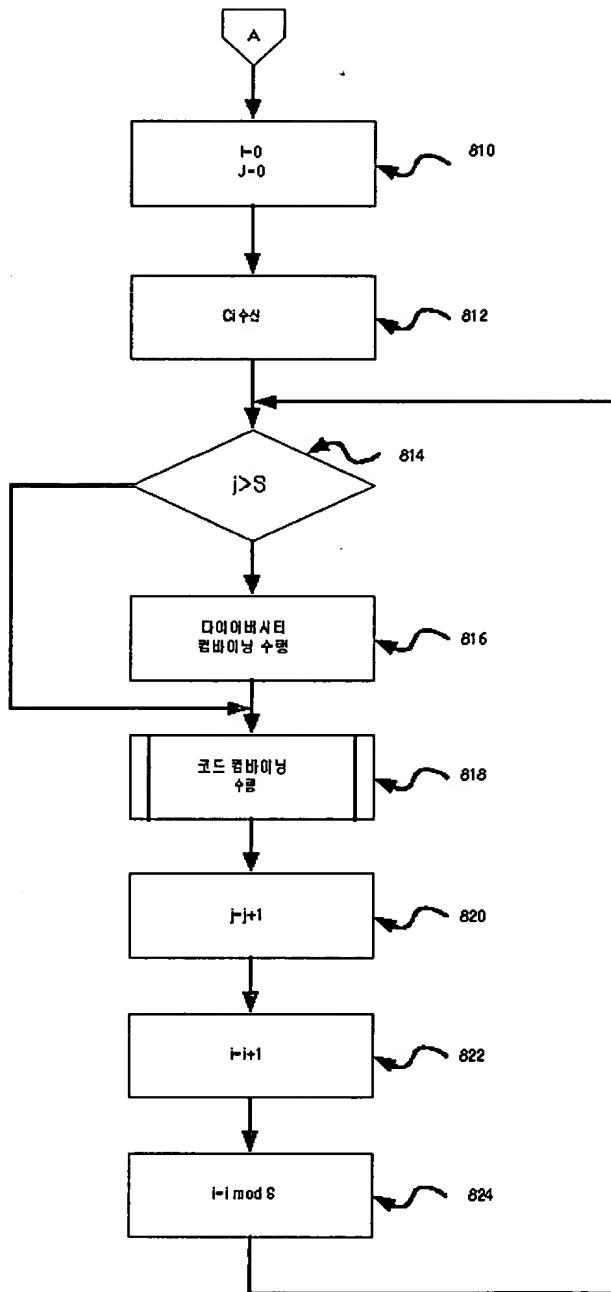
【도 9】



【도 10】



【도 11】



【도 12】

